IBM Agent Builder V 6.3.5

# 用户指南



**注** 在使用本资料及其支持的产品之前,请先阅读<u>第 329 页的『声明』</u>中的信息。

此版本适用于 IBM® Agent Builder V6.3.5 以及所有后续发行版和修订版,直至新版本中另有声明为止。

<sup>©</sup> Copyright International Business Machines Corporation 2010, 2017.

# 目录

<u></u>		.ix
表		xiii
第	1章 Agent Builder 概述	1
	公共 Agent Builder 过程	1
	数据源和数据集	2
	监视多个服务器或一个服务器的多个实例	
	测试。安装和配置代理程序	<u>د</u>
	是作系统重求	 Д
	特定于 IBM Tivoli Monitoring 的功能部件	5
第	2 章安装和启动 Agent Builder	7
-1-	安装和运行 Agent Builder 的先决条件	7
	Agent Builder 的详细系统需求	7
	安装 Agent Builder	/
	使田安华向导来安华 Agent Builder	/
	反用文表向与木文表 Agent Builder	, / Q
	时款又表 户动 Agent Builder	0
	后均 Agent Duilder 由沿罢钟少谢监察	9
	在 Agent Duilder 中汉里峡首冽见谷	10
	住 Agent Duilder 中反直畎首时间截に服労们码	. 10
	即戦 Agent Builder 静默卸载	. 10
第	3 章创建代埋桯序	.13
	对代理程序命名和配置	. 13
	定义初始数据源	14
	选择键属性	15
第	4 章使用 Agent Editor 修改代理程序	. 17
	缺省操作系统	18
	自描述代理程序	. 18
	环境变量	. 19
	环境变量列表	.19
	看守程序信息	26
	Cognos 信息	. 27
	"生成代理程序向导"链接	27
	" <b>数据源定义</b> "页面	27
	通过使用" <b>数据源定义</b> "页面来复制数据源	. 28
	" <b>运行时配置信息</b> "页面	. 28
	"代理程序 XMI 编辑器"页面	29
	保存编辑和更改	29
	家立代理程序的版 <b>术</b>	29
	为代理程序设置新版大号	ر <u>ح</u> م ک
	/ショ゙レヘーューュノジヒン旦タリリルやつ	. JU 20
第	5 章编辑数据源和特性属性	31
	创建、修改和删除属性	32
	创建属性	32

	复制属性	32
	编辑特性	32
	创建派生属性	. 33
	编辑派生属性	. 34
	移除属性	34
	用于定义特性的字段和洗项	
	尾性类型	36
	属性久工	36
	周位的奴了分面	
	指定周位的权平	20
	油足用仔癀心泪小狗的特性的一里住	. 50
	りに通用は组	. 30
	FOITINULA EUROI	. 39
	史以 Formula Editor 组针视图	39
	组件尖空	40
	Formula Editor 通用选坝	. 41
	Formula Editor - 公式错误	42
	公式运算符和函数	. 43
	指定操作系统	48
	11. 11. 11. 11. 11. 11. 11. 11. 11. 11.	48
	数据类型	49
第	6 章定义和测试数据源	53
• • •		54
	定义田干讲程浏览的连接	56
	とへ们了 近住/河辺り建设	56
	□ 1. Windows 派力 定\田干服冬浏览的许垤	57
	た又用」服另例処凹建设	J/ E0
	· 血恍木日 WINUOWS Management Instrumentation (WMI) 的奴据	30
	/则风WMI 周注组	. 59
	画税 WINDOWS 住舵曲税器 (Perrmon)	60
	测试 Pertmon 周注组	61
	监视米目间单网络官埕	61
	SNMP MIB 错误	. 64
	SNMP MIB 解析选项	64
	测试 SNMP 属性组	65
	监视来自简单网络管理协议事件发送方的事件	. 65
	SNMP 事件配置属性	. 67
	测试 SNMP 事件属性组	69
	监视 Java 管理扩展 (JMX) MBean	70
	JMX 配置	77
	JMX 通知	79
	JMX 监视器	. 79
	Java 管理扩展 (JMX) MBean 的特定字段	83
	测试 JMX 属性组	86
	监视来自公共信息模型 (CIM) 的数据	
	····································	00 87
	∭讨 ∩IM 届性组	88
	则成 GITT 属 正组	. 00
	四大山心大山	00
	山心又什麼你有力的心	
		90
	版 AIX 二进制日志	96 98
	) 监视 AIX 二进制日志 监视 Windows 事件日志	96 98 99
		96 98 99 100
		98 98 99 100 100
	<ul> <li>         協口応入(市)間は</li> <li>         监视 AIX 二进制日志         监视 Windows 事件日志         按事件类型过滤         按事件类型过滤         按事件源过滤         按事件标识过滤         按事件标识过滤         </li> </ul>	98 98 99 100 100 101
	<ul> <li>協口心へ市間は</li> <li>监视 AIX 二进制日志</li> <li>监视 Windows 事件日志</li> <li>按事件类型过滤</li> <li>按事件源过滤</li> <li>按事件标识过滤</li> <li>监视命令返回码</li> </ul>	98 98 100 100 101 101
	<ul> <li>協協協大市福住型</li> <li>监视 AIX 二进制日志</li> <li>监视 Windows 事件日志</li> <li>按事件类型过滤</li> <li>按事件源过滤</li> <li>按事件标识过滤</li> <li>监视命令返回码</li> <li>编辑命令文件定义</li> </ul>	98 98 100 100 101 101 104

从远程系统收集脚本数据	
脚本解析和分隔符	
监视脚本输出的步骤	
监视来自 Java 数据库连接 (JDBC) 的数据	109
LIDBC 配置	112
友储过程	113
到储定性	11 <i>1</i>
6 田 Ding 本收加系统可用性	۳±± 115
医用「IIIg 不皿%示乳可用は 和聖女研	
即且入计	
测试 PINg 周注组	
<u> </u>	
H11P 衣	
监视 URL	
监视 https:// URL	
代理服务器	
HTTP 配置	
测试 HTTP 属性组	
监视来自 SOAP 或其他 HTTP 数据源的数据	
JSON 数据的 XML 表示	
SOAP 属性的特定字段	
SOAP 配置	
测试 SQAP 属性组	130
使用套接字来监视数据	131
将在接字信息发送到代理程序	132
有接受为据的编码	135
至这了妖师吗编吗	136
	130
去按于癿且	
些性去按于蜥叫足按 在控宁的程 <del>术</del> 脚 <del>术</del>	120
丢汝子吵件平脚平	
测风县按子属性组	
使用 Java API 米監视数据	
运行 Java 应用程序	
生成的样本 Java 应用程序	
Java API 配直	
测试 Java 应用程序属性组	
第 7 章根据现有来源创建数据集	
连接两个属性组	
处理连接式属性组中的属性	154
连接式属性	
创建过滤属性组	
第8章创建导航器组	
お 7 早 (ア用ナ ア 只	
创建子节点	
子节点配置	
配置子节点	
子节点配置覆盖	
高级子节点配置	
从命令行配置子节点	
子节点配置示例	
子节点与 Windows 数据源	
子节点和脚本数据源	
第 10 章定制代理程序配置	175
使用 Agent Editor 更改配置属性	

	配置 Windows 远程连接	177
	创建具有 Windows Management Instrumentation (WMI) 权限的用户	
	配置安全 Shell (SSH) 远程连接	179
第		
• • •	创建情境、"执行操作"命令和杳询	
	创建工作空间	
第	12 章为 Cloud APM 准备代理程序	187
笛	13 章在 Agent Builder 中测试代理程序	
210		
	属性组测试 - 首选项	
	属性组测试 - 配置	
	代理程序全面测试	
	测试环境变量	197
竿	11 音在收初其础结构由安装代理程序以供测试和使用	100
בצ	工,早仁血化坐叫幻何个又表心生性广风层测风相区而	199
	本地安装代理程序	<u>1</u> 99
	创建代理程序包	
	在 IBM Tivoli Monitoring 环境中安装软件包	
	在 IBM Tivoli Monitoring 环境中配置和启动代理程序	
	在 IBM Cloud Application Performance Management 环境中安装和使用代理程序	
	生成及安装代理程序之后的结果	
	卸载代理程序	
	使用 Tivoli Enterprise Portal 移除 Tivoli Monitoring 代理程序	213
	在不使用 Tivoli Enterprise Portal 的情况下移除 Tivoli Monitoring 代理程序	214
	从 Tivoli Enterprise Portal 清除 Tivoli Monitoring 代理程序	
	卸载 IBM Cloud Application Performance Management 代理程序	214
第	15 章导入应用程序支持文件	
210	与出和导入 Tivoli Enterprise Monitoring Agent 的文件	
	导出和导入 Tivoli System Monitor Agent 的文件	
四	4/ 辛夷州社读和按西	240
퐈	10 早争计22 派和拘安	
	在 Tivoli Enterprise Portal 中查有事件过滤和摘要	
第	17 章故障诊断和支持	227
第附	17 章故障诊断和支持 录 Δ 共享项日文件	227 229
第 附	<b>17 章故障诊断和支持</b> <b>录 A 共享项目文件</b> 共享解决方案安装程序项目	227 229
第附	<b>17 章故障诊断和支持</b> 录 A 共享项目文件 共享解决方案安装程序项目	<b>227</b> <b>229</b> 229
第 附 附	<b>17 章故障诊断和支持</b> 录 A 共享项目文件 共享解决方案安装程序项目 录 B 命令行选项	227 229 229 
第 附 附	<b>17 章故障诊断和支持</b> <b>录 A 共享项目文件</b> 共享解决方案安装程序项目	
第 附 附	<b>17 章故障诊断和支持</b> <b>录 A 共享项目文件</b> 共享解决方案安装程序项目	
第附附	<b>17 章故障诊断和支持</b> <b>录 A 共享项目文件</b> 共享解决方案安装程序项目 <b>录 B 命令行选项</b> 命令 - generatelocal 命令 - generatemappingfile 命令 - generatezip	
第 附 附 附	<b>17 章故障诊断和支持</b>	
第 附 附 附	<b>17 章故障诊断和支持</b>	
第 附 附 附	<b>17 章故障诊断和支持</b> <b>录 A 共享项目文件</b> 共享解决方案安装程序项目	
第 附 附 附	<b>17 章故障诊断和支持</b>	
第 附 附 附	<b>17 章故障诊断和支持</b> 录 A 共享项目文件 共享解决方案安装程序项目	

"AIX 二进制日志"属性组	252
"监视器和通知"属性组	
计数器通知	256
标尺通知	259
已注册的监视器	
字符串通知	263
"SNMP 事件"属性组	
"JMX 事件"属性组	266
Ping 属性组	268
HTTP 属性组	270
发现属性组	275
Take Action Status 属性组	277
"日志文件状态"属性组	
"日志文件正则表达式统计信息"属性组	284
附录 D 为现有代理程序创建应用程序支持扩展	
创建"应用程序支持扩展"项目	
将支持文件添加至项目。	
生成 Application Support Extension 安装映像	
安装 Application Support Extension	290
将 <b>解决方案安装项目</b> 转换为"应用程序支持扩展"项目	291
附录 F 生成 Cognos 数据模型	203
<b>川 米 L 工成 Cognos 効用1天王</b> 生成 Cognos 物埕 構 刑 的 生 泣 久 件	•••••• <b>2/3</b> າດາ
主风 Cognos 数据候空的元/天来什 Tivoli Data Warabauca	293 202
Tivoli Data Walenouse	
Fremework Menader	
Framework Manager	
凹建妆亩 持衣 Mana da dCustana ≢	
現允 ManagedSystem 衣	
M Tivoli Common Reporting 中守山扳古和数据模型	
将扳方守入到 Agent Builder 中	
将代理程序包中的报告女装到 Tivoli Common Reporting 中	
附录 F ICU 正则表达式	
附录 G 创建非代理程序文件捆绑包	
远程部署捆绑包编辑器	
向捆绑包添加命令	
向捆绑包添加必备项	314
向捆绑包添加文件	
生成捆绑包	
为 Tivoli Netcool/OMNIbus 探测器创建可部署捆绑包	
附录 H 动态文件名支持	317
附录 I SNMP 陷阱配置	321
附录 1 "执行操作"命令关考	332
マンジン マート・アンジン マート・アンジン マート・アンジン マート・アンコン 2011 - 11 - 11 - 11 - 11 - 11 - 11 - 11	ອ <b>ບ</b> ອງເຊັ
辅助功能	327
声明	
	220

# <u>冬</u>

1. "进程监视器"页面示例5	55
2. "运行时配置"页面	8
3. 显示所收集 SNMP 事件数据的"测试事件设置"窗口	'0
4. "数据收集状态"窗口7	'0
5. JMX 连接属性7	'2
6. Java Management Extensions (JMX) Browser 窗口7	'4
7. JMX Agent-Wide Options 窗口7	'6
8. "添加过滤器"示例 1	)3
9. "添加过滤器"示例 29	<i>•</i> 4
10. 代理程序解析简单日志文件数据行时的示例属性值输出。	96
11. 代理程序解析复杂日志文件数据行时的示例属性值输出。	96
12. 显示已解析的日志文件属性值的"解析日志"窗口9	<del>)</del> 7
13. 代理程序解析复杂脚本输出时的属性值输出示例。	)6
14. "SOAP 浏览器"窗口12	25
15. "SOAP 浏览器"窗口12	26
16. "SOAP 浏览器"窗口12	27
17. 样本代理程序结构	2
18. "属性组信息"页面"属性组信息"窗口15	53
19. 找出源属性信息	55
20. 导航器树中的子节点16	0
21. 监视不同系统的子节点	51
22. 导航器树中的子节点类型	52
23. 监视相同子节点类型的多个子节点实例16	53

24. 示例:子节点中的数据收集	164
25. 展开的 SNMP V1 属性	168
26. Agent Builder 中的配置属性定义	
27. 包含 Agent Cfg 属性的代理程序级别配置的 Top 节	
28. 包含 Overridable Cfg 属性的代理程序范围缺省值的 Main 节	
29. 没有子节点的 Example Subnode 节页面	
30. 定义了两个子节点实例的 Example Subnode 节页面	173
31. 设置 sysadmin 用户标识	182
32. 设置 sysadmin 用户标识(续)	183
33. 设置 sysadmin 用户标识(续)	
34. 设置工作空间属性	185
35. 设置工作空间属性(续)	186
36. Agent Editor 的"代理程序信息"页面的"测试代理程序"部分。	194
37. 突出显示了示例子节点和导航器组的"代理程序测试"视图。	195
38. "代理程序测试"透视图	196
39. "属性组测试"视图,显示与 Managed_URLs 和 Managed_Nodes 属性组的数据收集有息 (Performance Object Status)。	ī关的更多信 197
40. "管理 Tivoli Enterprise Monitoring Services"窗口	209
41. 新代理程序中属性组的节点。	
42. "可用性"节点	
43. "性能对象状态"节点	
44. "事件日志"节点	213
45. 未启用事件过滤或摘要时的历史视图和高速缓存视图	221
46. 选择了仅发送摘要事件时的历史视图和高速缓存视图	222
47. 选择了发送所有事件时的历史视图和高速缓存视图	
48. 选择了发送第一个事件时的历史视图和高速缓存视图	224

49. 选择了事件阈值时的历史视图和高速缓存视图	225
50. 选择代理程序项目文件	298
51. 选择发布包	299
52. 选择 Common Reporting	
53. 选择 Report Studio	
54. Report Studio	
55. "内容管理"选项卡	
56. 列出代理程序包的"内容管理"选项卡	306
57. 第二和第三种类型的配置记录示例	



1. 创建代理程序的快速参考信息	1
2. 其他功能的快速参考信息	2
3. 环境变量	
4. 用于编辑数据源的字段	
5. 用于定义特性的字段和选项	35
6. 数字属性选项	
7. StringToTivoliTimestamp 的有效格式参数	
8. StringToTivoliTimestamp 示例	
9. "进程监视器"页面上的字段	54
10. SNMP 事件配置属性	
11. 过滤选项	
12. 支持用于监视代理程序的 SQL 数据类型	
13. "网络管理"配置属性	
14. 要在其中搜索所要监视的对象的 HTML 元素	
15. HTTP 属性信息 - 受管 URL	
16. HTTP 属性信息 - URL 对象	
17. URL 文件条目	
18."URL 监视"配置属性	
19. 代理服务器配置属性	
20. Java 配置属性	
21. SOAP 属性信息	
22. HTTP 服务器配置属性	
23. 补充文件的文件类型	

24. 样本错误代码	134
25. 属性值中要进行编码的字符	135
26. Performance Object Status 值	
27. 套接字配置属性	
28. 补充文件的文件类型	140
29. Java 跟踪级别选项	
30. 属性字段的数据类型及其 IBM Tivoli Monitoring 属性类型对等项	
31. 代理程序的内部错误码	
32. 要求对 Java 源代码进行修改的代理程序更改	
33. Java 配置属性	149
34. 源属性组 1(单行)	152
35. 源属性组 2(单行)	152
36. 生成的连接	152
37. 源属性组 1(单行)	152
38. 源属性组 2(多行)	152
39. 生成的连接	152
40. 源属性组 1(多行)	152
41. 源属性组 2(多行)	153
42. 生成的连接(按属性 3 和属性 7 连接)	
43. 环境变量	
44. 命令快速参照表	
45. 必需自变量	303
46. 可选自变量	303
47. 正则表达式元字符	309
48. 正则表达式运算符	

49. 替换文本字符	
50. 标志选项	
51. 命令的预定义变量	314
52. SNMP 数据提供程序支持的类别	322
53. SNMP 数据提供程序支持的严重性	
54. SNMP 数据提供程序支持的状态	323
55. SNMP 数据提供程序支持的源标识	

# 第1章 Agent Builder 概述

您可以使用 IBM Agent Builder 来创建和修改定制代理程序以扩展 IBM Tivoli<sup>®</sup> Monitoring 或 IBM Cloud Application Performance Management 环境的监视功能。定制代理程序使用上述任一环境来监视任何类型的内部或定制软件。

Agent Builder 基于 Eclipse,这是一种开放式源代码的集成开发环境。

Agent Builder 包含 Tivoli Monitoring 和 Cloud APM 环境的以下功能部件:

#### 定义和修改代理程序

您可以创建和修改代理程序。代理程序收集并分析不同资源(例如,磁盘、内存、处理器或应用程序)的状态和性能的相关数据,并将此数据提供给监视环境。

#### 测试和准备代理程序以用于部署

您可以在 Agent Builder 中测试代理程序, 方法是在 Agent Builder 运行所在的主机上收集数据(某些情况下也可以从其他主机上收集信息)。您可以将代理程序打包以便于分发和部署。

以下其他功能部件可用于 Tivoli Monitoring:

## 定制工作空间、情境和"执行操作"命令

您可以使用 Agent Builder 对附加工作空间、情境和"执行操作"命令进行打包,以作为在 Tivoli Monitoring 环境中运行的新代理程序或现有代理程序的应用程序支持扩展

### 报告数据模型

您可以使用 Agent Builder 来生成 Cognos<sup>®</sup> 数据模型,再用这些模型构建 Tivoli Common Reporting 报告。然后,可以将这些报告打包为代理程序映像的组成部分。

## 公共 Agent Builder 过程

下表列出可使用 Agent Builder 完成的主要过程。

您可以使用 Agent Builder 为 IBM Tivoli Monitoring 和 IBM Cloud Application Performance Management 环境创建代理程序。您还可以用来为 Tivoli Monitoring 环境创建应用程序支持扩展。通过创建工作空间和情境可创建应用程序支持扩展,从而增强一个或多个现有代理程序。

Agent Builder 必须先安装才能使用。有关指示信息,请参阅<u>第7页的『第2章 安装和启动 Agent</u> Builder』。

要创建、测试和使用代理程序,请按列示顺序完成下表中的过程。

表 1. 创建代理程序的快速参考信息

目标	请参阅	
使用"代理程序"向导创建代理程序。	· 第13页的『第3章 创建代理程序』	
为代理程序创建数据源和属性。 要点:对于 Cloud APM 环境,摘要仪表板最多可以显示 大约 5 个属性;其中一个属性必须表示代理程序或子节 点的总体状态。	· 第 31 页的『第 5 章 编辑数据源和特性属性』	
对于 Tivoli Monitoring 环境,请为代理程序创建工作空间和情境。 ·至少运行 Tivoli Monitoring V6.1 FP1 ·将 Tivoli Universal Agent 解决方案版本重置为"00" ·设置"AppTag"的值	<ul> <li>· 第 181 页的『第 11 章 创建工作空间、"执行操作" 命令和情境』</li> <li>· 第 217 页的『第 15 章 导入应用程序支持文件』</li> </ul>	

表 <i>1</i> . 创建代理程序的快速参考信息 <i>(</i> 续)			
目标	请参阅		
对于 Cloud APM 环境,请为代理程序创建资源定义和 仪表板。	·第 187 页的『第 12 章 为 Cloud APM 准备代理程序』		
对于 Tivoli Monitoring 环境,请为代理程序创建用于报告的 Cognos 数据模型。	· 第 293 页的『附录 E 生成 Cognos 数据模型』		
测试和调试所创建的代理程序,以确保监视信息的可用 性。	<ul> <li>· 第 191 页的『第 13 章 在 Agent Builder 中测试代理程序』</li> <li>· 第 231 页的『附录 B 命令行选项』</li> <li>· 第 17 页的『第 4 章 使用 Agent Editor 修改代理程序』.</li> </ul>		
生成安装包,并在受监视主机上安装代理程序。	· <u>第199页的『安装代理程序』</u>		
移除使用 Agent Builder 创建的代理程序。	·第 213 页的『卸载代理程序』		

您还可以使用 Agent Builder 来打包定制工作空间、情境和"执行操作"命令以作为现有代理程序的应用程序支持扩展。这些功能仅可用于 Tivoli Monitoring 环境:

表 2. 其他功能的快速参考信息				
目标	请参阅			
创建定制工作空间、情境和"执行操作"命令。	· <u>第 181 页的『第 11 章 创建工作空间、"执行操作"</u> 命令和情境』			
将应用程序支持扩展打包。	· 第 289 页的『附录 D 为现有代理程序创建应用程序支 持扩展』			
构建定制捆绑包。	· 第 313 页的『附录 G 创建非代理程序文件捆绑包』			

## 数据源和数据集

代理程序可以监视来自一个或多个数据源的信息。它以属性形式向监视基础结构呈现信息,属性组织为数据 集。

在创建代理程序时,必须为其定义数据源。您可以添加更多数据源。数据源定义代理程序收集监视信息的方式。

您可以使用 Agent Builder 来创建代理程序,以使用监视下列数据提供程序的信息的数据源:

- ·进程和服务可用性
- ·网络系统可用性(使用 ICMP Ping)
- ·命令返回码
- ·脚本输出
- ·Windows 事件日志
- $\cdot$  Windows Management Instrumentation (WMI)
- ·Windows 性能监视器 (Perfmon)
- ·简单网络管理协议 (SNMP)
- ·SNMP 事件

- ·超文本传输协议 (HTTP) 可用性和响应时间
- ·SOAP 或其他 HTTP 数据源
- · Java<sup>™</sup> 数据库连接 (JDBC)
- ·Java 应用程序编程接口 (API)
- ·Java 管理扩展 (JMX)
- ·公共信息模型 (CIM)
- ·日志文件
- ·AIX<sup>®</sup> 二进制日志
- ·套接字

您还可以使用其他开发工具来创建定制监视应用程序以通过日志、脚本输出和 Java API 数据源将信息传递 给代理程序。

在添加数据源时,Agent Builder 会将对应的数据集添加到代理程序。数据集对呈现给监视环境的信息进行组织。在 IBM Tivoli Monitoring 中,数据集称为属性组。

数据集由若干属性构成,这些属性是由数据源提供的值。监视环境每次查询代理程序时,都会从数据源访存 值,然后作为数据集中的属性返回这些值。

某些数据源可以在同一查询中返回若干行属性值(例如,如果数据源同时监视若干个服务)。

大多数数据源将信息呈现为一个数据集。SNMP 和 JMX 数据源根据配置可能会提供不同的信息集。为容纳这些信息,添加 SNMP 或 JMX 数据源时, Agent Builder 会创建多个数据集。

您可以编辑数据集来过滤数据和创建其他派生属性,即,使用公式根据现有属性计算而得的属性。您还可以 合并数据集,从而使用两个或两个以上数据集中的信息创建一个新数据集。这样,用户可以查看从不同数据 源合并的信息。

在 IBM Tivoli Monitoring 中,您可以查看所有属性内容。还可以创建工作空间,以定制视图呈现来自所有代理程序数据集的信息。您可以使用 IBM Tivoli Monitoring 来创建情境,当任何属性达到某个值时会触发这些情境。情境可以发出警报和调用系统命令。

在 IBM Cloud Application Performance Management 中,必须为代理程序定义摘要仪表板,最多可以选择 五个属性显示在该仪表板中。您也可以定义详细信息仪表板,以将来自任何数据集的信息显示为表。您可以 创建当任何属性达到某个值时触发的阈值;无需将此属性添加到仪表板。阈值可以发出警报。

## 监视多个服务器或一个服务器的多个实例

代理程序可以监视多个服务器,包括同一服务器的多个实例。可通过两种方法创建此类代理程序:代理程序的多个实例和代理程序中的若干子节点。

多个实例是监视应用程序服务器的标准方法,这些应用程序服务器在同一主机上拥有若干相似的实例。IBM Tivoli Monitoring 和 IBM Cloud Application Performance Management 中的许多标准代理程序支持多个实例。

通过多个实例,您可以在受监视主机上安装代理程序,然后为每个实例设置名称来配置一个或多个实例。为 要监视的每个服务器实例配置一个代理程序实例。每个实例都是代理程序的一个独立且相同的副本,可以单 独启动和停止。

您还可以在代理程序中定义一种或多种类型的子节点。每个类型所对应的资源类型都必须不同于代理程序能够监视的资源类型。子节点类型包含数据源和数据集;您还可以在任意子节点外部定义代理程序级别的数据源和数据集。在主机上安装代理程序时,可以对每种类型配置所需数量的子节点;对于每种子节点类型,可以独立设置子节点数量。对于 IBM Cloud Application Performance Management,您可以为代理程序创建仪表板,并且可以为每个子节点创建单独的仪表板。

子节点需要在受监视主机上执行不同的配置步骤。此外,要重新配置,需要添加或移除子节点。您必须停止 并重新启动整个代理程序;可以在不影响其他实例的情况下重新配置、添加或移除某个实例。但是,子节点 有若干优势:

- ·通过子节点,您可以在使用较少资源的情况下监视大量服务器实例。有一条准则是,单个系统上支持的特定类型的代理程序实例数量是10。但是,代理程序使用子节点最多可以监视100个本地或远程服务器。
- ·一个代理程序可以包含用于几种不同种类的服务器的子节点类型。在受监视系统上,您可以对每种类型配置任意数量的子节点。您可以使用此功能来进一步节省资源。
- ·使用子节点的代理程序可以在代理程序级别提供系统范围的数据。

您可以为同一代理程序同时定义多个实例和子节点。在此情况下,每个实例都可以包含若干子节点。您可以 独立于其他实例停止和重新启动每个实例;一个实例中的所有子节点会一起停止和重新启动。

## 测试、安装和配置代理程序

您可以为代理程序创建安装包,然后将其安装在任意数量的受监视主机上。对于某些数据源,需要设置配置 值来收集数据。

在为代理程序定义数据源和属性之后,可以在 Agent Builder 中运行该代理程序以进行测试。您可以测试单个数据集(属性组)或整个代理程序。

要扩大代理程序的测试范围并使用代理程序,可以创建安装映像。此映像提供了在任意受监视主机上安装和配置代理程序的脚本。

提示: 在安装代理程序之前,确保在主机上安装您的监视环境(IBM Tivoli Monitoring 或 IBM Cloud Application Performance Management)的操作系统代理程序。

在安装代理程序之后,您需要配置代理程序。如果代理程序支持多个实例,那么必须配置代理程序以创建至 少一个实例。

一些数据源需要额外的配置值;例如,对于 SNMP 数据源,必须使用 SNMP 协议配置所监视主机的 IP 地址。使用安装包部署的配置脚本来设置这些值。

或者,可以在创建安装映像之前,在 Agent Builder 中设置这些值。这样,您便无需在受监视主机上再次设置这些值。

提示:对 Cloud APM 服务器进行升级之后,定制代理程序的帮助文件可能不会显示在帮助内容中。要显示帮助文件,请完成下列步骤:

- 1. 下载最新版本的 IBM Agent Builder。
- 2. 重新创建定制代理程序。确保在"代理程序信息"页面中指定较高的版本号、修订包或补丁级别。
- 3. 将该定制代理程序安装在受监视主机上。
- 4. 在 Cloud APM 控制台中,单击导航栏中的帮助 > 帮助内容。这将显示您的定制代理程序帮助。

## 操作系统需求

Agent Builder 创建的代理程序在各种操作系统上受支持,是否受支持取决于监视环境以及创建代理程序时所选择的设置。

在 Tivoli Monitoring 环境中, Agent Builder 创建的代理程序可支持以下操作系统:

- $\cdot AIX$
- $\cdot$  HP-UX
- · Linux
- · Solaris
- $\cdot$  Windows

这些代理程序支持与操作系统代理程序相同的操作系统版本。有关详细信息,请访问<u>软件产品兼容性报告</u>Web站点。搜索Tivioli Monitoring产品名称,然后选择 OS Agents & TEMA (Tivoli Enterprise Monitoring Agent)组件复选框。

在 IBM Cloud Application Performance Management 环境中, Agent Builder 创建的代理程序可支持以下操 作系统:

 $\cdot AIX$ 

· Linux

· Windows

这些代理程序支持与操作系统代理程序相同的版本。有关详细信息,请使用 <u>System requirements (APM</u> <u>Developer Center</u>)的"组件报告"部分中的链接。

要在 Tivoli Monitoring 环境中运行监视代理程序,请在运行代理程序的每个受监视系统上安装相应的操作系统代理程序。

要在 IBM Cloud Application Performance Management 环境中运行监视代理程序,请在运行代理程序的每个受监视系统上安装 IBM Cloud Application Performance Management 随附的所有代理程序。

**注:** Agent Builder 浏览器对可从运行 Agent Builder 的系统访问的数据源和信息进行操作。请确保在以下一种系统上运行 Agent Builder:

·运行级别与您为其开发代理程序的操作系统和受监视应用程序相同的系统

·连接到另一个系统的系统,此另一个系统运行级别与您为其开发代理程序的操作系统和受监视应用程序相同

# 特定于 IBM Tivoli Monitoring 的功能部件

Agent Builder 提供了多个仅适用于 IBM Tivoli Monitoring 的功能部件。

您可以使用导航器组来组织代理程序显示在 IBM Tivoli Monitoring 导航器视图和工作空间中的数据。导航器 组将来自若干属性组(数据集)的数据组合到一个视图中,同时向用户隐藏原始独立数据集。

您可以使用 Tivoli Enterprise Portal 为代理程序创建工作空间、情境和"执行操作"命令。然后,可以使用 Agent Builder 将工作空间、情境和"执行操作"命令保存为应用程序支持文件,并将这些文件捆绑到代理程 序。此外, Agent Builder 还可以导入其他代理程序的工作空间、情境和"执行操作"命令,并为这些代理程 序创建定制应用程序支持文件。

Agent Builder 可以为代理程序生成 Cognos 数据模型。使用此数据模型将代理程序信息导入到 Cognos Framework Manager 中(这是 IBM Tivoli Common Reporting 的一部分)以创建报告。

# 第2章安装和启动 Agent Builder

在安装 IBM Agent Builder 之前,请确保系统满足先决条件。然后使用安装向导或静默安装过程来安装 Agent Builder。

提示: 有关安装或修改代理程序的信息, 请参阅 第 199 页的『安装代理程序』。

# 安装和运行 Agent Builder 的先决条件

要安装并运行 Agent Builder,您的系统必须符合特定的要求。

要安装 Agent Builder, 请确保满足以下条件:

- ·至少具有1GB可用磁盘空间的系统。您所开发的代理程序将需要额外磁盘空间。
- ·受支持的操作系统。Agent Builder 可运行以下操作系统:
- Windows
   Windows
- Linux (仅限 x86 64 位)
- · Linux 如果使用 Linux 操作系统,必须安装 libstdc++.so.5 库。您可以安装提供此库的以下包:
- 在 Red Hat Enterprise Linux 上为 compat-libstdc++-33
- 在 SUSE Enterprise Linux 上为 libstdc++-33

Windows 在 Windows 系统上,必须能够以具有 Administrator 许可权的用户身份运行 Agent Builder。这些 许可权可以确保 Agent Builder 的环境与用它开发的代理程序一致。

**Linux**在 Linux 系统上,可以 root 用户身份或普通用户身份运行 Agent Builder。但是,如果以普通用户 身份运行,代理程序测试将受到限制,并且在某些情况下可能不可用。

## Agent Builder 的详细系统需求

使用 Software Product Compatibility Reports 来查看 Agent Builder 的详细系统需求。

访问 软件产品兼容性报告 Web 站点。搜索 IBM Agent Builder 产品名称。

# 安装 Agent Builder

您可以使用安装向导或静默安装过程来安装 Agent Builder。

提示: 在安装 Agent Builder 之前,先卸载任何先前版本。有关卸载的更多信息,请参阅(<u>第10页的『卸</u>载 Agent Builder』)。卸载时不会丢失任何现有代理程序信息。

## 使用安装向导来安装 Agent Builder

您可以使用安装向导来安装 IBM Agent Builder。

## 开始之前

确保您的系统满足先决条件。有关先决条件的信息,请参阅<u>第7页的『安装和运行 Agent Builder 的先决条</u> 件』

### 过程

 如果您未登录到 <u>IBM Marketplace</u>,请使用您的 IBMid 和密码登录并转至"产品和服务"。
 "产品和服务"页面适用于活动订户。如果您有任何问题,请转至 <u>Cloud Application Performance</u> <u>Management 论坛或 Marketplace 支持</u>。

2. 下载 Agent Builder 安装归档文件:

- a) 在 Cloud APM 预订框中, 单击管理 > 下载。
- b) 选择**多平台**作为操作系统。
- c) 选择 IBM Agent Builder 程序包。
- d) 单击下载并将 IBM\_Agent\_Builder\_Install.tar 保存到系统。
- 3. 解压缩安装归档文件。
- 4. 在解压缩的映像目录中使用以下命令来启动安装:
  - · Windows setup.bat
  - · Linux UNIX ./setup.sh

要点:使用打算用来运行 Agent Builder 的相同用户标识来运行安装程序。

- 5. IBM Agent Builder 窗口打开后,请选择语言,然后单击确定。
- 6. 在"简介"页面上,单击下一步。
- 7. 在"软件许可证协议"页面上,单击我接受许可协议中的条款,然后单击下一步。
- 8. 在"选择安装文件夹"页面上,单击下列其中一个选项:
  - · 下一步, 用于将 Agent Builder 安装到您希望安装到什么位置? 字段中指定的目录。
  - · 复原缺省文件夹:用于将 Agent Builder 安装到缺省目录。
  - · 选择: 用于选择另一个目录。

注:所选目录名称不得包含以下字符:

- ! 非 %
- ;

如果包含其中任意字符,可能无法启动 Agent Builder。

9. 在"安装前摘要"页面上,单击安装。

10. 在"安装 IBM Agent Builder"页面上,等待安装完成页面打开,然后单击完成。

#### 结果

**Windows** 安装 Agent Builder 之后, "开始"菜单中会增加一个选项,并且桌面上会增加 Agent Builder 图 标。安装日志文件位于 *install\_dir*\IBM\_Agent\_Builder\_InstallLog.xml 中。

**Linux** 安装 Agent Builder 之后, Agent Builder 可执行文件命名为 *Install\_Location*/ agentbuilder。安装日志文件位于 *install\_dir*/IBM\_Agent\_Builder\_InstallLog.xml 中。

## 静默安装

您可以使用静默安装方法安装 Agent Builder。此方法不需要图形环境,可以在多台主机上轻松复制。

## 关于此任务

静默安装选项文件 installer.properties 包含在安装映像上的安装目录根目录中。您必须修改此文件 以满足您的需要,然后运行静默安装程序。您可以将此文件复制到其他主机,并将 Agent Builder 快速安装 到所有这些主机上。

### 过程

- 如果您未登录到 <u>IBM Marketplace</u>,请使用您的 IBMid 和密码登录并转至"产品和服务"。
   "产品和服务"页面适用于活动订户。如果您有任何问题,请转至 <u>Cloud Application Performance</u> <u>Management</u>论坛或 <u>Marketplace</u> 支持。
- 2. 下载 Agent Builder 安装归档文件:a) 在 Cloud APM 预订框中,单击管理 > 下载。

- b) 选择**多平台**作为操作系统。
- c) 选择 IBM Agent Builder 程序包。
- d) 单击下载并将 IBM\_Agent\_Builder\_Install.tar 保存到系统。
- 3. 解压缩安装归档文件。
- 4. 创建 installer.properties 文件的副本,该文件位于安装映像目录中。
- 5. 编辑新文件以满足您的需要。以下是该文件的内容示例:

```
ŧ
# IBM Agent Builder
∃Ŀ
# (C) Copyright IBM Corporation 2009. All rights reserved.
# Sample response file for silent install
# To use this file, use the following command:
#
# Windows:
    setup.bat -i silent -f <path>\installer.properties
#
1
# Linux or AIX:
    setup.sh -i silent -f <path>/installer.properties
‡ŧ
#
# Where
‡ŧ
     <path> is a fully-quailfied path to the installer.properties
     file (including the drive letter or UNC path name on Windows).
#
ŧ
    <path> cannot contain spaces.
#
# -
                                     # This property indicates that the license has been accepted
# LICENSE ACCEPTED=FALSE
                                              # This property specifies the install directory
1
# On Windows, the default is:
      C:\\Program Files (x86)\\IBM\\AgentBuilder
#
# On Linux, the default is:
# /opt/ibm/AgentBuilder
±
#USER_INSTALL_DIR=C:\\Program Files (x86)\\IBM\\AgentBuilder
#USER_INSTALL_DIR=/opt/ibm/AgentBuilder
```

6. 通过在解压缩的安装映像目录中运行以下命令来启动静默安装:

Windows setup.bat -i silent -f path/installer.properties

Linux \_\_\_\_\_./setup.sh -i silent -f path/installer.properties

其中, *path* 是 installer.properties 文件的标准路径(在 Windows 上包括盘符或 UNC 路径名)。路径不得包含空格。

## 启动 Agent Builder

安装 Agent Builder 之后,可以将其启动。

## 过程

- · 使用下列其中一种方法来启动 Agent Builder
  - · Windows 在 Windows 系统上:
    - 从命令行输入: Install\_Location\agentbuilder.exe。
    - 选择开始 > 所有程序 > IBM > Agent Builder。
    - 单击 Agent Builder 桌面图标。

在 Linux 系统上, 启动以下可执行文件: INSTALL\_DIR/agentbuilder

**注:** 运行 Agent Builder 时,它会提示您输入工作空间目录的位置。用于创建代理程序的文件保存在该目录中。您可将任何目录指定为工作空间。

# 在 Agent Builder 中设置缺省浏览器

Linux 在 Linux 系统上,您可能需要设置 Agent Builder 缺省浏览器,以便显示帮助面板。

#### 过程

- 1. 选择窗口 > 首选项以打开"首选项"窗口。
- 2. 选择并展开常规节点。
- 3. 选择 Web 浏览器。
- 4. 选择使用外部 Web 浏览器。
- 5. 选择要使用的浏览器。
- 6. 可选: 要添加 Web 浏览器, 请完成下列步骤
  - a) 单击新建。
  - b) 在名称字段中, 输入浏览器的描述性名称。
  - c) 在位置字段中, 输入浏览器可执行文件的完整路径。
  - d) 单击**确定**。
- 7. 单击确定。

## 在 Agent Builder 中设置缺省时间戳记服务机构

您可以在 Agent Builder 首选项窗口中为 JAR 文件设置时间戳记服务机构。如果缺省时间戳记服务机构签名 证书到期,通过设置新服务机构,您可以继续验证 JAR 文件。

## 过程

- 1. 选择窗口 > 首选项以打开"首选项"窗口。
- 2. 选择并展开 IBM Agent Builder 节点。
- 3. 选择 JAR 签名。
- 4. 选择将时间戳记添加到已签名的 JAR 文件。
- 5. 输入时间戳记服务机构的 URL。
- 6. 单击确定。

## 卸载 Agent Builder

根据您的操作系统,可以使用不同的过程来卸载 Agent Builder。

## 过程

## Linux

- 在 Linux 系统上,运行以下命令:
- a) INSTALL\_DIR/uninstall/uninstaller

其中 INSTALL\_DIR 是安装 Agent Builder 的目录的名称。

Windows

在 Windows 7、Windows Server 2008 R2 和更高版本的 Windows 上,请完成以下步骤: a) 选择**开始 > 控制面板 > 程序 > 程序和功能**,打开 "Windows 程序和功能"。

- b) 从已安装程序的列表中选择 IBM Agent Builder。
- c) 单击卸载/更改。
- d) 在"卸载 IBM Agent Builder"页面上,单击卸载。
- e) 在"卸载完成"页面上, 单击完成。

**提示:** 在 Windows 7 和 Windows Server 2008 R2 上,您还可以通过选择**开始 > 计算机 > 卸载或更改程序** 来转至"Windows 程序和功能"窗口。然后,从步骤 <u>第 11 页的『2』</u>继续操作。

## Windows

- 在其他 Windows 系统上,完成以下步骤:
- a)从 Windows 控制面板,选择添加/删除程序。
- b) 单击 IBM Agent Builder。

c) 单击更改/删除。

- · 在所有操作系统上,您还可以使用静默卸载方法。通过运行以下命令启动静默卸载:
  - · Windows 在 Windows 系统上: INSTALL\_DIR/uninstall/uninstaller.exe -i silent
  - · Linux 在 Linux 系统上: INSTALL\_DIR/uninstall/uninstaller -i silent

## 静默卸载

您可以使用静默卸载方法执行卸载。

## 过程

· 通过运行以下命令启动静默卸载:

INSTALL\_DIR/uninstall/uninstaller[.exe] -i silent

# 第3章创建代理程序

要开始在 Agent Builder 中创建代理程序,请使用"新建代理程序"向导。通过此向导,您可以设置基本代理程序配置并创建一个数据源。然后,可以在 Agent Builder 中处理该代理程序以添加更多数据源和其他选项,包括子节点和导航器组。

## 对代理程序命名和配置

使用"代理程序"向导对代理程序命名,设置代理程序版本、受支持的操作系统和其他配置设置。

### 过程

- 1. 使用下列其中一种方法来启动"新建代理程序"向导:
  - a) 单击工具栏上的 影 创建新代理程序图标。
  - b) 从主菜单中,选择文件 > 新建 > 代理程序。
  - c) 从主菜单中,选择**文件 > 新建 > 其他**。在"选择向导"页面中,双击 Agent Builder 文件夹,然后双 击**代理程序**。

"代理程序"向导将打开。

- 2. 单击下一步。
- 3. 在"新建代理程序项目"页面中的项目名称字段中,设置项目名称。Agent Builder 将此名称用于包含代 理程序文件的文件夹。您可以选择更改以下设置:
  - ·如果要将代理程序文件存储在其他位置,请取消选中**使用缺省位置**,并单击**浏览**以在**位置**字段中选择 新目录。
  - 通过将资源添加到各种工作集,可更改 Eclipse Navigator View 显示资源的方式。有关更多信息,请 参阅 Eclipse 帮助。要向 Eclipse 工作集添加代理程序,请选择将项目添加至工作集并单击选择按钮, 将工作集添加到工作集字段。
- 4. 单击下一步。
- 5. 在"常规信息"页面中, 配置以下设置:
  - · 在版权字段中输入要用于新代理程序的版权声明。该声明必须符合对版权的法律要求。此版权声明会 插入到为代理程序生成的所有文件中; 您之后可以对其进行编辑。
  - · 选择要为其构建代理程序的操作系统。

**要点:** 如果想要在 Agent Builder 内部对代理程序运行全面测试(有关指示信息,请参阅<u>第 194 页的</u> <u>『代理程序全面测试』</u>),请确保:

- 如果在 Windows 上运行 Agent Builder, 安装 32 位版本的操作系统。
- 如果在 Linux 上运行 Agent Builder, 安装 64 位版本的操作系统。

**要点:** 在极少数情况下,您可能需要在仅安装有 32 位操作系统代理程序的 64 位系统上安装代理程 序。在此情况下,请确保不要选择 64 位版本的操作系统,而选择 32 位版本。

**要点:**使用 Agent Builder 创建的代理程序不支持 64 位 Windows Server 2003 R2 及更早版本的 Windows 系统。

- 6. 单击下一步。
- 7. 在"代理程序信息"页面中, 配置以下设置:
  - 在服务名称字段中,为代理程序设置服务名称。此名称会显示在 IBM Tivoli Monitoring 环境中的"管理 Tivoli Monitoring Services"窗口以及 IBM Cloud Application Performance Management 中的 "管理 Monitoring Services"实用程序和阈值编辑器中。在 Windows 系统上,这也是运行代理程序的 Windows 服务的名称。完整的服务名称始终以 Monitoring Agent for 开头。您负责输入名称

的余下部分,这个部分通常用来描述此代理程序所监视的服务。名称可以包含字母、数字、空格和下 划线。

 · 在产品代码字段中为代理程序设置三字符产品代码。IBM Tivoli Monitoring 和 IBM Cloud Application Performance Management 都需要产品代码。已保留一系列产品代码以用于 Agent Builder。允许的值 为 K00-K99、K{0-2}{A-Z} 和 K{4-9}{A-Z}。

要点:这些值仅供内部使用,而不用于要在组织外部共享或销售的代理程序。如果要创建与他人共享的代理程序,必须向 toolkit@us.ibm.com 发送说明,以保留产品代码。请求产品代码必须包含要构建的代理程序的描述。然后,将为您分配、注册和返回产品代码。当您收到三个字母的产品代码时,系统将告知您如何使 Agent Builder 能够使用分配的产品代码。

- · 在**公司标识**字段中设置一个字符串,用于唯一标识开发代理程序的组织(IBM 是保留字)。您可以从 公司的 URL 获取该标识;例如,如果公司 Web 站点是 mycompany.com,可使用文本 mycompany。
- · 在**代理程序标识**字段中设置一个字符串以唯一标识代理程序。在缺省情况下, Agent Builder 将代理程 序标识设置为与产品代码相同。

要点:代理程序标识字段和公司标识字段的长度之和不能超过11个字符。

· 在版本字段中设置代理程序版本。代理程序版本以 V.R.R 格式包含一个三位数,其中:

V = 版本

R=发行版

R = 发行版

为了在监视环境中显示, V.R.R 值会转换为以下格式: 0V.RR.00.00

提示: 在 Agent Editor 中,提供了**补丁级别**字段。在为代理程序发布修订时,可以使用**补丁级别**字段,而无需更新版本。

 · 如果希望代理程序支持多个实例,请选中支持此代理程序的多个实例复选框。您可以使用代理程序的 多个实例来监视同一主机上某个应用程序的若干实例,或者使用一个主机上安装的代理程序来监视不 同主机上的若干软件服务器。在安装代理程序以支持多个实例时,可以创建所需数量的实例并进行配置。

#### 下一步做什么

单击下一步,为代理程序定义初始数据源。有关更多信息,请参阅<u>第14页的『定义初始数据源』</u>

## 定义初始数据源

在创建代理程序时,请定义该代理程序要监视的初始数据。稍后,您可以在 Agent Editor 中添加更多数据源。

关于此任务

使用"代理程序初始数据源"页面来定义新代理程序要监视的数据源。有关根据各种数据提供程序创建数据 源的详细指示信息,请参阅<u>第53页的『第6章定义和测试数据源』</u>。

#### 过程

1. 在"代理程序初始数据源"页面上,选择其中一个监视数据类别及其中一个数据源。

2. 单击下一步。向导将引导您完成定义和配置任何所指定数据收集类型的过程。

**提示:**您可以使用此向导来定义数据源,或者添加子节点或导航器组以组织代理程序。有关子节点的更多 信息,请参阅<u>第 159 页的『第 9 章 使用子节点』</u>。有关仅用于 IBM Tivoli Monitoring 的导航器组的更多 信息,请参阅<u>第 157 页的『第 8 章 创建导航器组』</u>。

- 如果您已定义可以返回多个数据行的新数据源,那么将提示您选择键属性。有关更多信息,请参阅<u>第</u> 15页的『选择键属性』。
- 4. 在定义首个数据源之后,将显示"数据源定义"窗口。要添加另一个数据源,请选择代理程序,或者选择子节点或导航器组(如果存在),然后单击添加到选定项按钮。
- 5. 要完成定义数据源,请单击完成。Agent Builder 会创建新代理程序并在 Agent Editor 中将其打开。

## 选择键属性

属性组返回多个数据行时,您必须选择键属性。

## 关于此任务

属性组可以返回多个数据行时,每一行代表一个正受监视的实体。每次对受监视数据采样时,监视环境都将 行与受监视实体以及该实体的先前采样匹配。此匹配可通过键属性来完成。属性组中的一个或多个属性可以 标识为键属性。这些键属性组合起来可以区分受监视实体。当键属性表示同一受监视实体时,不会从一个样 本切换到下一个样本。

速率和变化量属性通过比较当前样本与先前样本计算得出。键属性相同可确保代理程序正在比较同一受监视 实体的值。同样, Summarization and Pruning Agent 汇总具有相同键属性的样本。另外,在某种情境下, 任何设置为键属性的属性还可以用作"显示项"。

您可以在"代理程序初始数据源"页面中指定新数据源的相关详细信息。如果所选数据源可能会返回多个数据行,那么 Agent Builder 有时可能会检测到键属性。否则,会提示您选择键属性。

### 过程

- · 在"选择键属性"页面上,请执行下列其中一个步骤:
  - ·从列表中单击作为该实体的键属性的一个或多个属性。要选择多个属性,请按住 Ctrl 键。
  - ·如果此属性组仅返回一行,请选择**生成单一数据行**。如果选择了该选项,那么无需任何键属性,原因 是该属性组中只会报告一个受监视实体。

# 第4章使用 Agent Editor 修改代理程序

使用 Agent Editor 更改、保存和落实代理程序的版本。

您可以在 Agent Builder 中创建新代理程序;有关更多信息,请参阅<u>第 13 页的『第 3 章 创建代理程序』</u>。 创建代理程序之后,可以使用 Agent Editor 修改此代理程序。

要在 Agent Editor 中打开您在 Agent Builder 中创建的代理程序,可在 Project Explorer 窗格中,找到该代 理程序的名称并将其展开。在该代理程序名称下面,双击**代理程序定义**。或者,双击 itm\_toolkit\_agent.xml 文件名。

Agent Editor 是包含多个页面的 Eclipse 编辑器,您可以用于修改现有代理程序的属性。编辑器中的每个页面都对应于代理程序的一个特定功能。

可用页面列表显示在"大纲"视图中的**代理程序定义**节点下方。您可以通过在"大纲"视图中单击节点轻松 切换到其他页面。如果没有"大纲"视图,或者此视图隐藏在另一视图下方,可以重置"代理程序定义"透 视图。选择**窗口 > 重置透视图**可重置此透视图。另外,也可以右键单击**代理程序定义**选项卡,并从菜单中选 择**重置**。

注: 有关创建代理程序的详细信息和过程, 请参阅第 13 页的『第 3 章 创建代理程序』。

Agent Editor 中包含以下页面:

- · 第17页的『"代理程序信息"页面』
- · "数据源定义"页面
- · "运行时配置信息"页面
- · \_ "代理程序 XML 编辑器"页面 (itm\_toolkit\_agent.xml)

**注:** 在查看 Editor 页面时,还可以通过单击其他页面的选项卡来切换到该页面。某些页面只有在"大纲"视 图中选中后才显示选项卡。您可以强制使页面在未选中时显示选项卡。要强制使页面显示选项卡,请单击针 形图标以使该图标中的针形指向该页面。

#### "代理程序信息"页面

"代理程序信息"页面是 Agent Editor 的主页。

"代理程序信息"页面包含以下信息:

·一般代理程序信息,其中包括代理程序服务名称和产品代码。您可以单击**高级**,以设置不同的名称用于不同用途,但通常不需要进行此设置。

代理程序内容信息

- 缺省操作系统链接
- 自描述代理程序链接
- 环境变量链接
- 看守程序信息链接
- Cognos 信息链接
- 数据源链接
- 运行时配置链接
- 资源链接
- 仪表板链接
- · 测试代理程序链接
- ·生成代理程序向导链接
- · 落实代理程序版本链接

## 配置瞬时错误消息的时间

Agent Editor 向导有时显示瞬时错误消息。消息会在向导标题中显示片刻(缺省情况下显示 3 秒)。您可以 配置显示这些消息的持续时间。要更改此设置:

- 1. 从 Agent Builder 菜单栏中选择窗口 > 首选项。"首选项"窗口随即打开。
- 2. 选择 Agent Builder。
- 3. 设置瞬时错误消息显示的时间(秒)设置。
- 4. 单击确定。

# 缺省操作系统

使用缺省操作系统页面可更改构建代理程序时针对的操作系统。

#### 过程

- · 要打开缺省操作系统页面,请单击"代理程序信息"页面的"代理程序内容"部分中的缺省操作系统, 或者单击"大纲"视图中的缺省操作系统节点。
- · 在缺省操作系统页面中,选择您的代理程序一定可以支持的操作系统。

在为代理程序生成安装程序包时, Agent Builder 会将所选操作系统的文件添加到该程序包。添加到代理 程序而且并非特定于 Windows 操作系统的数据源在选择的所有操作系统上都可用。您可以更改任何特定 数据源所适用于的操作系统,使其不同于此缺省选择。要更改可用于特定数据源的操作系统,请使用 "数据源定义"页面的"操作系统"窗格。如果未选择缺省操作系统,那么您必须在"数据源定义"页 面上针对每一个特定数据源选择操作系统。

**要点:** 如果想要在 Agent Builder 内部对代理程序运行全面测试(有关指示信息,请参阅<u>第 194 页的『代</u> 理程序全面测试』),请确保:

- 如果在 Windows 上运行 Agent Builder, 安装 32 位版本的操作系统。
- 如果在 Linux 上运行 Agent Builder, 安装 64 位版本的操作系统。

**要点:** 在极少数情况下,您可能需要在仅安装有 32 位操作系统代理程序的 64 位系统上安装代理程序。在此情况下,请确保不要选择 64 位版本的操作系统,而选择 32 位版本。

## 自描述代理程序

对于 IBM Tivoli Monitoring 环境,请使用"自描述代理程序"页面来指定是否将代理程序的支持文件与代理 程序捆绑在一起。对于 IBM Cloud Application Performance Management 环境,必须保持启用自描述代理 程序。

#### 过程

· 要打开"自描述代理程序"页面,请单击"代理程序信息"页面的"代理程序内容"部分中的自描述代 理程序,或单击"大纲"视图中的自描述代理程序节点。

缺省情况下,针对使用 Agent Builder 6.2.3 或更高版本创建的所有新代理程序启用自描述。如果代理程序用于 IBM Cloud Application Performance Management 环境,必须启用自描述。

对代理程序启用自描述功能后,应用程序支持包将包括在代理程序映像中。包括在内将使代理程序能够 植入 Tivoli Enterprise Monitoring Server、Tivoli Enterprise Portal Server 和 Tivoli Enterprise Portal Browser 的支持文件。有关自描述代理程序的更多信息,请参阅《IBM Tivoli Monitoring 安装与设置指 南》和《IBM Tivoli Monitoring 管理员指南》。在 IBM Cloud Application Performance Management 环 境中,自描述使代理程序能够将支持文件植入到 Cloud APM 服务器上;植入过程是此环境中的必需步 骤。

注: 在 IBM Tivoli Monitoring 环境中,必须安装 Tivoli Monitoring V6.2.3 或更高版本才能使自描述代理程序功能工作,并且必须在 Tivoli Monitoring 中启用自描述。缺省情况下,自描述功能在 Tivoli Monitoring 中处于关闭状态。

注:选中针对此代理程序启用自描述复选框不会阻止代理程序在先前版本的 Tivoli Monitoring 上工作。

## 环境变量

使用"环境变量"页面杳看和修改代理程序在运行期间可以使用的环境变量。

### 开始之前

有关 "Agent Editor" 和 "代理程序信息"页面的更多信息,请参阅<u>第 17 页的『第 4 章 使用 Agent Editor</u> 修改代理程序』。

### 关于此任务

环境变量可以是您在脚本中为访问权定义的变量,也可以是使代理程序按照某种方法运行的预定义变量。请 参阅第19页的『环境变量列表』以获取预定义变量列表。

### 过程

- 1. 要打开"环境变量"页面,请在"代理程序信息"页面的代理程序内容部分中单击环境变量。或者,单 击"大纲"视图中的环境变量节点。
- 2. 在"环境变量"页面中,单击添加以添加新变量。或者,要编辑现有变量,请将其选中并单击编辑。
- 3. 在"环境变量信息"窗口中,设置以下值:
  - · 在**名称**字段中, 输入变量名称, 或者从列表中选择预定义名称。
  - · 在值字段中,如果要为代理程序设置变量,输入变量值。如果不输入值,代理程序会传播现有变量的 值。
  - · 在描述字段中, 输入变量描述, 或者保留预定义变量的现有描述。
  - a) 单击确定。 新增变量列示在"代理程序信息"页面的表中。

## 环境变量列表

**—** 

使用环境变量来控制代理程序运行时行为。

通过使用"环境变量"页面,可以将环境变量构建到代理程序中。在 Windows 系统上,环境变量在代理程 序的 KXXENV 文件中定义。在 UNIX 和 Linux 系统上,可以在代理程序的 \$CANDLEHOME/config/XX.ini 文件中定义这些变量。XX 是两字母产品代码。代理程序必须重新启动才能使新设置生效。

注: 未在运行 C Shell 的远程系统上正确设置环境变量。如果要使用环境变量,请使用其他 shell。

表 3. 环境变量. 下表列出了环境变量、缺省值、有效值范围以及每个变量的描述					
环境变量	缺省值	有效值	描述		
CDP_ATTRIBUTE_GROUP_ REFRESH_INTERVAL	不适用	任何非负整 数	在后台更新指定属性组的时间间隔 (秒)。此变量工作方式与 CDP_DP_REFRESH_INTERVAL 相同,除 了其目标仅为指定属性组外。变量名中的 属性组名称必须为大写,即便实际属性组 名称不为大写也是如此。		
CDP_DP_CACHE_TTL	55	大于等于1 的任何整 数。	在此秒数内,对所收集的属性组数据进行 高速缓存。在该时间间隔内,对相同数据 的多次请求将收到该数据的高速缓存副 本。该值应用到代理程序中的所有属性 组。		

表 3. 环境变量. 下表列出了环境变量、缺省值、有效值范围以及每个变量的描述 (续)						
环境变量	缺省值	有效值	描述			
CDP_ATTRIBUTE_GROUP_CACHE_ TTL	CDP_DP_CACHE _TTL 的值	大于等于1 的任何整 数。	在此秒数内,对所收集的指定属性组数据 进行高速缓存。在该时间间隔内,对相同 数据的多次请求将收到该数据的高速缓存 副本。此值覆盖指定组的 CDP_DP_CACHE_TTL。变量名中的属性组 名称必须为大写,即便实际属性组名称不 为大写也是如此。			
CDP_DP_IMPATIENT_ COLLECTOR_TIMEOUT	如果定义了子节 点则为 5, 否则 不设置	任何正整数	在超时和返回高速缓存数据前(即使高速 缓存数据是旧数据)等待数据收集的秒 数。(如果高速缓存数据存在时间超过 CDP_DP_CACHE_TTL秒,那么为旧数 据)。如果未设置此变量,代理程序将等 待,直到数据收集完成。等待有时会使 Tivoli Enterprise Portal 超时并放弃等待。 如果未配置任何线程池,将忽略该变量并 将同步进行数据收集。			
CDP_DP_REFRESH_INTERVAL	如果定义了子节 点则为 60,否则 不设置	任何非负整 数	在后台更新属性组的时间间隔(秒)。如 果该变量未设置或者设置为0,将禁用后 台更新。如果配置了线程池(请参阅变量 CDP_DP_THREAD_POOL_SIZE),那么 可以用并行方式刷新属性组。如果没有线 程池,那么将按顺序方式进行更新,这可 能会花费很长时间。在逻辑上等同于线程 池大小为1。			
CDP_DP_THREAD_POOL_SIZE	如果定义了子节 点则为 15, 否则 不设置	任何非负整数	为按照 CDP_DP_REFRESH_INTERVAL 定 义的时间间隔运行后台数据收集而创建的 线程数。如果该变量未设置或者设置为 0,表示没有线程池。 如果 CDP_DP_THREAD_POOL_SIZE 设置 为大于 1 的值,且 CDP_DP_REFRESH_INTERVAL 设置为 0,那么将忽略 CDP_DP_THREAD_POOL_SIZE 的值并按 需进行数据收集。 Thread Pool Status 属性组显示线程池是 如何运行的。使用 Thread Pool Status 来 调整线程池大小和刷新时间间隔以获得最 佳结果。缺省情况下,该属性组的查询不 会在代理程序导航器树上显示。您可能会 忘记在代理程序的定制工作空间中包含查 询。但是,可以通过将 Thread Pool Status 查询指定到基本代理程序级别工作 空间视图进行轻松查看。			
CDP_JDBC_MAX_ROWS	1000	任何正整数	JDBC 数据提供程序返回的最大数据行数。对于包含的行数超过该数量的结果集,最多仅处理此最大数量的数据行。可开发查询以避免将太多数据返回到 IBM Tivoli Monitoring。			
表 3. 环境变量. 下表列出了环境变量、缺省值、有效值范围以及每个变量的描述 (续)						
---	---------------	------------------------	--			
环境变量	缺省值	有效值	描述			
CDP_NT_EVENT_LOG_GET_ALL _ENTRIES_FIRST_TIME	NO	YES 或 NO	如果设置为 YES,那么代理程序将针对 Windows 事件日志中的每个事件发送一个 事件。如果设置为 NO,那么仅发送 Windows 事件日志中的新事件。			
CDP_NT_EVENT_LOG_CACHE _TIMEOUT	3600	大于等于 300 的任何 整数。	代理程序对 Windows 事件日志事件进行高 速缓存的秒数。当查询事件日志属性组 时,将返回所有高速缓存的事件。 注:不再使用该变量。而使用 CDP_PURE_EVENT_CACHE_SIZE 变量。			
CDP_PURE_EVENT_CACHE_SIZE	100	大于等于1 的任何正整 数。	针对以下项高速缓存的最大事件数:配置 为处理新记录的日志文件数据源、 Windows 事件日志属性组。以及 JMX 监 视器和通知。日志中的每个新记录都将导 致发送事件。该环境变量定义代理程序在 高速缓存中记住的事件数。当查询该属性 组时,将返回高速缓存值。			
CDP_DP_ACTION_TIMEOUT	20 秒	大于等于1 的任何正整 数。	等待代理程序正在处理的"执行操作"完成的秒数。			
CDP_DP_SCRIPT_TIMEOUT	30 秒	大于等于10 的任何正整 数。	等待基于脚本的属性组所启动程序完成的 秒数。			
CDP_DP_PING_TIMEOUT	30 秒	大于等于10 的任何正整 数。	等待命令返回码启动的程序完成的秒数。 注:此变量与 ICMP Ping 数据提供程序无 关。			
CDP_SNMP_MAX_RETRIES	2	任何正整数	发送 SNMP 请求重试次数。如果未收到响应,那么发送到 SNMP 代理程序的请求总数是该值加一。			
CDP_SNMP_RESPONSE_TIMEOUT	2秒	任何正整数	等待每个 SNMP 请求超时的秒数。属性组 中的每行表示一个单独的请求。此超时值 是在进行重试前等待响应的秒数。单行数 据的总超时为 (CDP_SNMP_MAX_RETRIES + 1) * CDP_SNMP_RESPONSE_TIMEOUT。总的 缺省超时值是 (2+1) * 2 = 6 秒。			
CDP_DP_HOSTNAME	安装的第一个网络接口的名称	IP 地址或主 机名	在多接口系统上设置首选主机名(网络接 口)。如果代理程序将其侦听端口绑定到 非缺省网络接口地址,那么请使用此环境 变量。此环境变量由 SNMP 数据提供程序 使用。 对于套接字数据源,如果同时设置了 CDP_DP_ALLOW_REMOTE,那么此变量也 适用。			

表 3. 环境变量. 下表列出了环境变量、缺省值、有效值范围以及每个变量的描述 (续)			
环境变量	缺省值	有效值	描述
CDP_SNMP_ALLOW_ DECREASING_OIDS	NO	YES 或 NO	如果设置为 YES, SNMP 数据提供程序不 会检查返回的 OID 是否在增大。设置为 YES 时需小心,因为受监视代理程序可能 有问题,通常此检查会捕获这些问题。
KUMP_DP_COPY_MODE_SAMPLE_I NTERVAL	60	等待时间 (秒)	对于日志文件数据提供程序,在将代理程 序定义为 <b>对文件进行采样时处理所有记录</b> 时,指定其重新读取文件内容前的等待时 间。以秒为单位指定此时间。
KUMP_MAXPROCESS	100%	5-100%	对于日志文件数据提供程序,指定要用来 处理文件数据的最大处理器使用率。值的 范围为 5-100%。缺省值为 100%。
KUMP_DP_SAMPLE_FACTOR	5	任何非负整 数	对于日志文件数据提供程序,在 Agent Builder 上选择 <b>对文件进行采样时处理所有</b> 记录时,设置采样因子。此等待时间确保 在针对模式记录扫描前写入跨多个记录的 模式。
KUMP_DP_EVENT	5	任何非负整 数	对于日志文件数据提供程序,设置事件数 据的采样频率(秒)。
KUMP_DP_FILE_EXIST_WAIT	YES	YES 或 NO	对于日志文件数据提供程序,指定文件监视线程检测到受监视的文件不存在或者为空,该线程继续运行。该线程将等待文件出现,每隔几秒钟重新检查一次,并在文件可用时启动或重新启动监视。
KUMP_DP_FILE_SWITCH_ CHECK_INTERVAL	600	任何非负整 数	以秒为单位的频率,当启用动态文件名支 持时,日志文件数据提供程序按此频率搜 索要切换到的其他监视文件。
KUMP_DP_FILE_ROW_ PAUSE_INCREMENT	无	任何非负整 数	对于日志文件数据提供程序,指定在文件 监视线程暂停前读取的文件记录数。暂停 文件监视线程是以便处理之前的更新。仅 当受监视文件突然收到大量新记录,并且 您担心可能会丢失某些记录更新时,才使 用此环境变量。
CDP_COLLECTION_TIMEOUT	60 秒	任何正整数	代理程序等待另一个进程启动的数据收集 器做出响应的秒数。示例包括 JMX、 JDBC、HTTP 和 SOAP 数据收集器。
CDP_SSH_TEMP_DIRECTORY	. (句点)	远程系统上 的任何有效 路径字符串	对于启用 SSH 的脚本数据提供程序,指定 远程系统上的位置。随代理程序提供的脚 本文件将更新至此位置。相对位置相对于 用户的主目录。缺省值.(句点)表示用 户的主目录。
CDP_SSH_DEL_COMMAND	rm -Rf	远程系统上 的任何有效 删除命令字 符串	对于启用 SSH 的脚本数据提供程序,指定 命令以开始删除代理程序随附的已上载脚 本文件。

表 3. 环境变量. 下表列出了环境变量、缺省值、有效值范围以及每个变量的描述 (续)			
环境变量	缺省值	有效值	描述
CDP_SNMP_SEND_DELAY_ FACTOR	0 毫秒	任何正整数	SNMP 初始发送从 0 延迟为指定的毫秒数。仅当同时启用了线程池时才会启用此变量。该延迟不会应用到所有发送,而只会应用到属性组进行的第一次发送。当受监视设备由于同时收到多个请求而难以做出正确响应时,此变量非常有用。
CDP_ICMP_PING_REFRESH_ INTERVAL	60 秒	大于等于1 的任何整数	在此时间间隔内已经对设备列表文件中的 系统执行了 Ping 操作。如果 Ping 使用太 长时间,那么再次开始 Ping 之前,总是至 少延迟 CDP_PING_MIN_INTERVAL_DELAY 秒。 数据的刷新频率不会超过此设置。可以根 据设备列表文件中的条目数量和接收响应 所用时间来降低刷新数据的频率。
CDP_ICMP_PING_MIN_ INTERVAL_DELAY	30 秒	大于等于 <b>1</b> 且小于 CDP Ping 刷新时 间间隔的任 何整数	在对设备列表文件中的设备执行 Ping 之 后,下一个 Ping 刷新时间间隔至少要在此 秒数之后才会开始。
CDP_ICMP_PING_BURST	10	大于等于 0 的任何整数	在代理程序暂停时间达到 CDP_ICMP_PING_BURST_DELAY 变量指 定的值之前发送的 Ping 数。值为 0 将禁用 此功能。
CDP_ICMP_PING_BURST_DELAY	10	大于等于 0 的任何整数	在发送 CDP_ICMP_PING_BURST 变量所 定义数量的一组 Ping 之后等待的时间量 (毫秒)。值为 0 将禁用此功能。
CDP_ICMP_PING_TIMEOUT	2000 毫秒	大于等于1的任何整数	等待 Ping 响应的毫秒数。此设置适用于所做的每次 Ping 尝试。系统会对每个主机尝试 3 次 Ping 操作。如果 3 次尝试都未收到响应,那么等待应答的总时间为 CDP_ICMP_PING_TIMEOUT 乘以 3。在缺省情况下,此值为 6000 毫秒。更改 CDP_ICMP_PING_TIMEOUT 的值将导致当前响应时间属性的缺省 TIMEOUT 枚举 不再适用。将 TIMEOUT 枚举更改为新值,即 CDP_ICMP_PING_TIMEOUT 乘以 3。
CDP_JDBC_CONNECTIONLESS	false	true 或 false	如果设置为 true,那么将在每次数据收集 尝试后关闭 JDBC 连接。即,所有属性组 都会在每次进行数据收集时尝试创建自己 的连接。如果启用了此变量,那么不会复 用连接。如果设置为 false,那么建立与数 据库的一个连接后,属性组之间将共享该 连接。
CDP_SSH_EXCLUDED_ ENVIRONMENT_VARIABLES	无	以逗号分隔 的环境变量 名称列表	对于启用了 SSH 的脚本数据提供程序,指 定不允许在远程系统环境中设置的一组本 地环境变量。

表 3. 环境变量. 下表列出了环境变量、缺省值、有效值范围以及每个变量的描述 (续)			
环境变量	缺省值	有效值	描述
CDP_DP_EVENT_LOG_MAX_ BACKLOG_TIME	0秒	0、1 或大于 1 的任何整 数	如果设置为 0, 并且 CDP_DP_EVENT_LOG_MAX_BACKLOG_EV ENTS 未设置为 1 或更大的整数, 那么不 处理在代理程序关闭期间生成的事件。0 是缺省值。
			如果设置为 1,并且 CDP_DP_EVENT_LOG_MAX_BACKLOG_EV ENTS 未设置为大于 1 的整数,那么将处 理在代理程序关闭期间生成的所有事件。
			如果设置为大于1,并且 CDP_DP_EVENT_LOG_MAX_BACKLOG_EV ENTS 未设置为大于1,那么将处理以当前 计算机时间为准,该秒数值内生成的事 件。例如,如果值设置为300,那么在启 动时,代理程序处理当前时间300秒内生 成的所有事件。
			在针对 CDP_DP_EVENT_LOG_MAX_BACKLOG _TIME和 CDP_DP_EVENT_LOG_MAX _BACKLOG_EVENTS 变量输入大于1的值 时,将处理此时间间隔内的事件或此数量 的事件。选择哪个变量取决于哪一项最先 匹配。
CDP_DP_EVENT_LOG_ Windows_Event_Log_MAX_BACK LOG_ TIME	0秒(不处理代 理程序关闭期间 错过的事件)	0、1或大于 1的任何整 数	如果设置为

表 3. 环境变量. 下表列出了环境变量、缺省值、有效值范围以及每个变量的描述 (续)			
环境变量	缺省值	有效值	描述
CDP_DP_EVENT_LOG_ MAX_BACKLOG_EVENTS	0个事件	0、1或大于 1的任何整 数	如果设置为 0, 并且 CDP_DP_EVENT_LOG_MAX_BACKLOG _TIME 变量未设置为 1 或更大的整数,不 处理在代理程序关闭期间生成的事件。0 是缺省值。
			如果设置为1,并且 CDP_DP_EVENT_LOG_MAX_BACKLOG _TIME 变量未设置为大于1的整数,那么 将处理在代理程序关闭期间生成的所有事 件。
			如果设置为大于 1, 并且 CDP_DP_EVENT_LOG_MAX_BACKLOG _TIME 未设置为大于 1, 那么最多将处理 在代理程序关闭期间生成的此数量的事 件。例如, 如果值设置为 200, 那么在代 理程序启动时, 将处理启动前生成的最近 200 个事件。
			在针对 CDP_DP_EVENT_LOG_MAX_BACKLOG _EVENTS 和 CDP_DP_EVENT_LOG_MAX_BACKLOG _TIME 输入大于 1 的值时,将处理此数量 的事件或此时间间隔内的事件。选择哪个 变量取决于哪一项最先匹配。
CDP_DP_EVENT_LOG_ Windows_Event_Log_MAX_BACK LOG_ EVENTS	<ul> <li>0 个事件(不处 理代理程序关闭 期间错过的事 件)</li> </ul>	0 或者大于 等于 1 的任 何整数	如果设置为
CDP_HTTP_READ_TIMEOUT	10	任何正整数	等待对 HTTP 请求的应答的秒数。
CDP_JAT_THREAD_POOL_SIZE	15	任何正整数	Java 提供者为处理数据收集请求而使用的 线程数。JMX、JDBC、HTTP 和 SOAP 数 据提供程序是可受益于此线程池的提供 者。
CDP_HTML_OBJECTS_THREAD_ POOL_SIZE	10	任何正整数	这是一个线程数,这些线程用于下载在使用 HTTP 数据提供程序监视的 URL 中找到的页面对象。
CDP_HTTP_SOAP_MAX_ROWS	500	任何正整数	HTTP SOAP 数据提供程序返回的最大行数。
CDP_DP_ALLOW_REMOTE	NO	NO 和 YES	如果设置为 YES,代理程序将允许远程套接字连接。如果设置为 NO,代理程序将只允许来自本地主机的套接字连接。NO是缺省值。
CDP_DP_INITIAL_COLLECTION_ DELAY	不固定	任何正整数	代理程序启动之后,线程池开始其调度的 数据收集之前的秒数。

# 看守程序信息

使用"看守程序信息"页面以指定代理程序看守程序的配置信息。

### 关于此任务

要打开"看守程序信息"页面,请在"代理程序信息"页面的代理程序内容部分中单击看守程序信息。您还可以选择"大纲"视图中的看守程序信息节点。

可以为代理程序看守程序指定以下配置信息:

#### ·缺省情况下监视此代理程序

选中此复选框以在安装该代理程序后将代理程序交由 Agent Management Services 管理。将监视代理程序 是否有不正常行为或异常终止,并且由看守程序重新启动。

#### ·检查频率(秒)

看守程序检查代理程序进程是否存在任何不正常行为或异常终止的频率。缺省值为每180秒一次。

#### ·最大重新启动次数

在向管理员发出问题警报前,在 24 小时周期内由于发生不正常行为或异常终止,看守程序重新启动代理 程序的次数。该周期是从每天的午夜开始。因此,代理程序启动的第一个周期可能会"很短"。

如果代理程序因任何原因关闭,将发生重新启动。如果因代理程序无反应或运行不正常,那么看守程序也 会停止。例如,超出内存阈值时。缺省设置是在 24 小时周期内重新启动四次,该周期为从午夜到晚上 11:59。代理程序的每日重新启动计数在午夜自动归零。

### ·内存阈值信息

这是代理程序进程的大小上限(以兆字节计),如果代理程序大小增大到超出此值,那么其看守程序会认定它运行不正常。Windows、Linux和 UNIX 各有不同的值。如果代理程序进程大小超过此阈值,看守程序将停止并重新启动该进程。这些属性没有缺省值。如果不指定值,看守程序不会监视进程大小。此度量在 Windows 上使用工作集大小,在 UNIX 和 Linux 上使用用户内存大小。

如果看守程序停止代理程序,并且到达最大重新启动次数,那么看守程序发送代理程序超过其重新启动计数 的警报,并停止执行自动重新启动。看守程序仍将报告代理程序是启动还是停止(假定其以另一种方式启 动,例如,通过 Tivoli Enterprise Portal)。

必须使用 AMS 启动代理程序"执行操作"命令手动重新启动代理程序,以使重新启动计数不被重置。

此计数在以下某种情况下会重置(看守程序继续工作并报告状态,但不执行自动重新启动):

·时钟指向午夜12点。

·用户使用 AMS 启动代理程序"执行操作"命令,此命令具有名为 resetRestartCount 的输入参数。如果 输入值 1(表示"true"或"yes"),那么每日重新启动计数将重置为 0。

有关更多信息,请参阅 IBM Tivoli Monitoring Administrator's Guide 中的以下部分:

·对于 Tivoli System Monitor Agent

Configuring Agent Management Services on Tivoli System Monitor Agents

·对于 Tivoli Enterprise Monitoring Agent

Installing and configuring Tivoli Agent Management Services

# Cognos 信息

使用"**Cognos 信息**"页面来指定为代理程序生成 Cognos 数据模型时使用的信息。此信息仅用于 IBM Tivoli Monitoring 环境。

### 过程

- 1. 要打开"Cognos 信息"页面,请单击"代理程序信息"页面的代理程序内容部分中的 Cognos 信息,或 单击"大纲"视图中的 Cognos 信息节点。
- 2. 在**数据源**字段中,输入将 Tivoli Common Reporting 连接至 IBM Tivoli Data Warehouse 的数据源的名称。

缺省值为 TDW。

3. 在模式字段中,输入用于 Tivoli Data Warehouse 的数据库模式的名称,此名称用来对 Cognos 报告中的 表名进行完全限定。

缺省值为 ITMUSER。将生成的 Cognos 模型装入 Framework Manager 时,可以在 Framework Manager 中更改此值。

"数据源定义"页面中的向报告类别添加此属性组复选框确定此属性组在 Cognos 模型中的放置位置。如果未选中此复选框,则会将此属性组放置在 Cognos 模型内的扩展属性文件夹中。如果选中此框,那么此属性组将放入"关键度量"文件夹中的所选子文件夹("可用性"或"性能")。有关数据源字段的更多信息,请参阅第 31 页的表 4。

### 下一步做什么

您可以使用 Cognos 数据模型为代理程序创建 Tivoli Common Reporting 报告,请参阅 <u>第 293 页的『附录 E</u> <u>生成 Cognos 数据模型』</u>。

# "生成代理程序向导"链接

创建或编辑新的代理程序后,请使用"生成代理程序向导"准备安装。

过程

· 完成创建或编辑新代理程序之后,在 Agent Editor 的"代理程序信息"页面上,单击生成代理程序向导链接。

通过"生成代理程序向导",您可以:

- 在本地系统上使用 Tivoli Monitoring 安装生成代理程序文件。有关指示信息,请参阅<u>第199页的『本</u><u>地安装代理程序』</u>。
- 创建程序包以便可以在其他系统上安装该代理程序。有关指示信息,请参阅<u>第 201 页的『创建代理程</u> <u>序包』</u>。

# "数据源定义"页面

使用"数据源定义"页面来操作数据源。

### 关于此任务

"数据源定义"页面列出针对代理程序配置的数据源。当您在树中选择某个数据源或特性时,此页面将更新为显示选定对象的属性。使用这些字段可修改所选数据源或特性的属性。

**注:** 有关根据各种数据提供程序创建数据源的详细指示信息,请参阅<u>第53页的『第6章定义和测试数据</u> 源』。 过程

- · 要打开"数据源定义"页面,请单击"代理程序信息"页面的"代理程序内容"部分中的数据源,或单击"大纲"视图中的数据源节点。
- · 通过单击添加到选定项或者在导航树中单击右键并选择其中一个选项, 可以添加更多数据源。
- · 通过右键单击数据源和特性并选择移除, 可以将其移除。
- · 您可以添加、修改和移除特性。有关指示信息,请参阅第 31 页的『第 5 章 编辑数据源和特性属性』

### 通过使用"数据源定义"页面来复制数据源

使用"数据源定义"页面来复制数据源。

#### 开始之前

转至"数据源定义"页面。有关更多信息,请参阅<u>第 27 页的『"数据源定义"页面』</u>

#### 关于此任务

您可以将产生属性组的数据源复制到剪贴板,然后粘贴回到此代理程序或者粘贴到另一代理程序。不产生属性组的数据源包括"可用性"及"Windows事件日志"数据源。

#### 过程

1. 选择要复制的属性组。

2. 使用下列其中一种方法剪切或复制属性组:

- · 单击菜单栏中的编辑 > 剪切 > 编辑 > 复制。
- · 右键单击其中一个已选择的项目, 然后单击菜单中的剪切或复制。
- · 使用其中一个将会调用剪切或复制操作的操作系统或 Eclipse 击键。例如,在 Windows 系统上,按下 Ctrl-C 将调用复制操作。

要将数据源从现有位置移除并放置到剪贴板中,请使用**剪切**。要将数据源保留在原位置并复制到剪贴板,请使用**复制**。

- 3. 选择属性组的父代(代理程序、子节点或导航器组),或选择现有属性组。
- 4. 通过使用下列其中一个选项, 粘贴选择内容:
  - · 从菜单栏中选择编辑 > 粘贴。
  - · 在树中, 右键单击要粘贴选择内容的节点, 然后单击菜单中的粘贴。
  - · 使用其中一个将会调用粘贴操作的操作系统或 Eclipse 击键。例如,在 Windows 系统上,按下 Ctrl-V 将调用粘贴操作。

#### 结果

剪贴板中的属性组将放置在选择的父代中。或者,如果已选中属性组,那么剪贴板中的属性组将放置在选定属性组的父代中。

如果粘贴时与另一属性组发生名称冲突,那么会对粘贴的属性组名称稍作更改,以避免冲突。

## "运行时配置信息"页面

"运行时配置信息"页面显示代理程序中的可配置变量。在受监视主机上安装代理程序时,您可以为变量设置值。

这些值可用于整个环境中的命令返回码和脚本。要打开"运行时配置信息"页面,请单击"代理程序信息"页面的"代理程序内容"部分中的运行时配置,或单击"大纲"视图中的运行时配置节点。Agent Builder 通过产品代码和标签自动构造环境变量的名称。

您可以使用"运行时配置信息"页面来添加和更改配置属性以及提供缺省值。

# "代理程序 XML 编辑器"页面

"代理程序 XML 编辑器"页面显示代理程序定义的 XML。

代理程序定义 XML 包含 Agent Builder 所有其他部分中显示的信息。如果更改 XML, 那么 Agent Builder 中显示的信息也会反映此更改。



**注意:** 请勿在 XML 中执行任何更改。此类更改可能导致错误,阻止生成代理程序或者对代理程序运行 步 造成负面影响。

# 保存编辑和更改

在保存使用编辑器所做的更改之前,不会存储这些更改。

#### 过程

· 通过以下某种方法来执行保存:

·选择文件 > 保存,这相当于选择保存(软盘)图标。

·按Ctrl+S。

在保存时,系统将进行验证以确保信息完整。如果发生问题,那么 Eclipse 的"问题"视图中将显示关于错误的信息。如果此视图不可见,请选择窗口 > 显示视图 > 问题。如果尝试生成的代理程序有错误,将显示错误消息。

注:必须先更正所有错误并保存更改,然后才能生成和安装代理程序。

# 落实代理程序的版本

如果您确定已完成了此版本的代理程序的开发,并且已准备好交付,请落实代理程序。

#### 关于此任务

IBM Tivoli Monitoring 系统要求新版本代理程序包含之前在监视环境中使用的该代理程序先前版本中包含的 所有信息。如果新代理程序安装在一些受监视主机上,但旧代理程序保留在其他主机上,那么必须包含先前 版本的所有信息,工作空间、情境和查询才能够继续工作。

在完成代理程序开发和测试之后,必须将代理程序落实为某个版本号的最终版本。Agent Builder 确保在落实 代理程序之后不会移除信息。后续构建代理程序时会使用新版本号。

最多只能有 1024 个版本。

**切记:** 如果您更改了要在 IBM Cloud Application Performance Management 环境中测试和运行的代理程序, 则必须更改代理程序版本。

#### 过程

1. 打开 Agent Editor 窗口的"代理程序信息"页面。

2. 在落实代理程序版本区域中, 单击落实此级别。

3. 备份已落实的代理程序,或者将其检入到版本控制系统中。

### 下一步做什么

落实代理程序后,对代理程序的其他任何更改都成为新版本的一部分。您必须先输入新版本号,然后才能保存其他更改。对新版本的任何更改均不得中断与先前版本代理程序的兼容性。

落实代理程序后,不能对落实该代理程序之前就存在的对象执行下列操作:

·从属性组中删除属性。

·删除属性组。

- ·对属性组中的现有属性重新排序。
- ·重新组织现有属性组(使用导航器项)。
- ·将属性组或导航器组移入或移出子节点。
- ·重命名属性组。
- ·重命名属性。
- ·更改现有属性的数据类型。
- ·更改子节点名称或类型(如果其中包含在落实代理程序前就存在的属性组)。
- ·更改代理程序的公司标识或代理程序标识。
- ·更改代理程序的产品代码。有关更多信息,请参阅<u>第30页的『更改产品代码』</u>。
- 在落实代理程序后,可以执行下列操作:
- ·将新属性添加到现有属性组。
- ·添加新属性组。
- ·对新属性重新排序。
- ·使用导航器项组织新属性组。
- ·创建新的子节点类型。
- ·添加新查询。
- ·添加新情境。
- ·添加新工作空间。

# 为代理程序设置新版本号

要保存对已落实的代理程序的更改,您必须输入新的版本号。

#### 过程

- 1. 打开 Agent Editor 窗口的"代理程序信息"页面。
- 2. 在"版本"提示后输入高于当前级别的版本、修订路径或补丁级别。
- 3. 对代理程序进行编辑。

提示:如果您落实了代理程序但忘记更改代理程序版本,系统会在您保存任何更改时提示您输入新版本。

# 更改产品代码

如果更改产品代码,那么您的代理程序将与该代理程序的任何先前版本都不兼容。先前落实操作的所有记录都会丢失,并且您将开发新的代理程序。

任何从 IBM Tivoli Monitoring 中导出并导入到代理程序中的文件、情境、"执行操作"命令或工作空间都会从该代理程序中删除。

如果您尝试更改已落实的代理程序的产品代码,那么 Agent Builder 会显示警告并询问您是否要继续。

在"**代理程序产品代码**"窗口中单击**是**之后,系统会显示警告,指出代理程序支持文件的内容不再有效。系统还将发出警告,指出下一次保存代理程序时将移除这些文件。

# 第5章编辑数据源和特性属性

在向代理程序添加数据源时, Agent Builder 会创建对应的数据集。您可以编辑数据集和其中的特性以提供必要的监视信息。

### 过程

要编辑或移除数据集(属性组)中的信息:

- 1. 在"代理程序信息"页面的"代理程序内容"区域中,单击数据源。 "数据源定义"页面将打开。
- 2. 选择数据集(属性组)。

该页面的属性组信息区域将更新以显示所选数据集的属性。

**注:** 或者,如果您处于"**代理程序**"向导的最后一页,则可双击数据源以打开"**属性组信息**"窗口。此窗口包含与"**数据源定义**"页面的属性组信息区域相同的信息。

第31页的表4描述了适用于所有数据源的字段信息。使用这些字段可修改所选数据源或特性的属性。

表 4. 用于编辑数据源的字段			
描述	可接受的值和示例		
Tivoli Enterprise Portal 或 IBM Cloud Application Performance Management 控制台中显示的数 据源的名称	可接受的值:长度小于或等于 32 个字符的描述性字符串。它在代 理程序中必须唯一。第一个字符 必须是字母,其余字符可以是字 母、数字或下划线。下划线显示 为空格。请勿使用空格或特殊字 符。		
数据源的帮助文本	可接受的值:最多 256 个字符的 字符串。		
数据源返回1行数据。在所有采 样数据源中均可编辑。	示例:如果要监视物理系统内 存,请选择单行。系统通常在单 个池中管理其所有内存;因此只 能返回一行数据。		
该数据源可以返回任意数量的数 据行。在所有采样数据源中均可 编辑。	示例:如果要监视磁盘驱动器, 请选择多个行,因为系统中可能 有多个磁盘。对于键,请选择可 区分磁盘的特性。对于磁盘,键 特性是盘号、盘符、卷标或您环 境中适用的任何特性。		
数据源返回基于事件的数据,每 个事件返回1行数据。	示例:超出性能阈值时,基于 SNMP 事件的数据源发送通知 (陷阱)。 注:并非所有数据源都能够生成事		
	描述Tivoli Enterprise Portal 或 IBM Cloud Application Performance Management 控制台中显示的数据源的名称数据源的帮助文本数据源区回1行数据。在所有采 样数据源中均可编辑。该数据源可以返回任意数量的数据行。在所有采样数据源中均可编辑。该数据源可以返回任意数量的数据行。在所有采样数据源中均可编辑。		

表 4. 用于编辑数据源的字段 (续)		
字段名称	描述	可接受的值和示例
将此特性添加至报告类别	这是生成的 Cognos 模型中,将 此属性组中的特性分配到的类 别。	如果选中此复选框,那么此属性 组将放入 Key Metrics 文件夹中 的所选子文件夹(Availability 或 Performance)。如果未选中此 复选框,那么属性组会放置在 Cognos 数据模型中的 <b>扩展度量</b> 文 件夹中。
度量类别	这是此属性组中的特性将分配到 的类别。	请选择 <b>性能</b> 或可用性。

注:

- a. 生成单一数据行和可生成多个数据行字段不影响事件数据源的数据。
- b. 有关已采样和事件数据类型的更多信息,请参阅<u>第49页的『数据类型』</u>。
- c. 有关特定数据源字段的信息,请参阅<u>第53页的『第6章定义和测试数据源』</u>中的相关数据提供程序信息。

## 创建、修改和删除属性

您可以在数据集(属性组)中创建、修改或删除属性。

要处理属性,请打开"数据源定义"页面。有关更多信息,请参阅<u>第 27 页的『"数据源定义"页面』</u>。

### 创建属性

您可以向数据集添加新属性。

#### 过程

右键单击该数据源,然后从菜单中选择添加属性。
 "属性信息"页面随即显示。

注:显示的页面取决于属性的数据源。

2. 在"属性信息"页面上指定新属性的选项。

有关这些字段及选项的信息,请参阅第34页的『用于定义特性的字段和选项』。

- 3. 要添加更多属性,请选择添加其他属性,并单击下一步。
- 4. 完成添加属性时,请单击完成。

### 复制属性

可以从"数据源定义"页面中复制属性。

### 过程

1. 在 Agent Editor 的"数据源定义"页面中,右键单击要复制的属性,然后单击复制属性。

2. 在"复制属性"窗口中的名称字段中,输入新属性的名称,然后单击确定。

### 编辑特性

您可以使用"数据源定义"页面编辑和更改特性信息。

### 过程

1. 选择要编辑的特性。

该页面的"特性信息"窗格将更新以显示选定特性的属性。

2. 指定对新特性信息的选择。

**注:** 在"**代理程序**"向导的最后一页("**数据源定义**"页面)上,可以双击特性以打开"**特性信息**"窗口。该窗口包含与"**数据源定义**"页面的"特性信息"窗格相同的信息。

### 创建派生属性

您可以创建从其他属性派生值(而不是直接从数据源派生值)的属性。

### 关于此任务

在派生的属性中,可对源属性的值执行运算。例如,对数字属性执行基本的算术运算或对字符串属性执行字 符串并置。

用于派生表达式的基本表达式语法包含函数。这些函数提供了较复杂的数据处理,包括短期聚集、从字符串 到整数的转换以及访问配置属性和环境变量。另外,还提供了一个编辑器,可帮助您在构建表达式时进行显示。

### 过程

1. 在"数据源定义"页面上,右键单击数据源,然后单击添加属性。

- 2. 在"属性信息"页面上, 输入属性名称和帮助文本。
- 3. 选择从其他属性值派生

4. 在公式字段中,输入公式文本,或者单击编辑以使用图形编辑器来输入公式。

有关可以在公式中使用的运算符及函数的信息,请参阅第43页的『公式运算符和函数』。

**注:** 当您单击**编辑**时,公式编辑器将打开。有关编辑派生属性的信息,请参阅<u>第 34 页的『编辑派生属</u>性』。

- 5. 可选: 选中或取消选中**特定于区间的计算**复选框,以确定计算函数时要使用哪两个属性样本值。 如果您的公式使用了 rate 或 delta 函数,请使用此选项。有关**特定于区间的计算**的更多信息,请参阅 <u>第 33 页的『特定于区间的计算』</u>。有关 rate 和 delta 函数的更多信息,请参阅<u>第 43 页的『公式运</u> <u>算符和函数』</u>。
- 6. 在**属性类型**区域中,单击属性类型。
- 7. 单击确定。

数据源定义页面将再次显示,并像以前一样列出数据源。

8. 单击**完成**。

要点:如果您所创建的派生属性要引用另一个派生属性,请确保所引用的属性在新属性之前列出。如果属性引用在清单中稍后位置列出的另一个派生属性,那么代理程序将无法显示此属性的值。如果您创建这样的属性,那么 Agent Builder 将显示警告。

### 特定于区间的计算

在定义基于 rate 或 delta 函数的派生属性时,可以选择特定于区间的计算。

您可在"**属性信息**"页面的"**派生属性详细信息**"选项卡上选择**特定于区间的计算**。有关更多信息,请参阅 <u>第 33 页的『创建派生属性』</u>。

在使用**特定于区间的计算**选项时,了解变化量(属性值之间的差值)概念很重要。变化量是指属性最新值和 属性先前值的差值。变化量由 delta 函数直接返回,并由 rate 函数用于计算结果。

delta或 rate函数必须始终以 last函数作为其唯一自变量。last函数指定使用属性的哪些值来确定变化量。如果未选择特定于区间的计算,则使用的先前值始终是第二新的值。如果选择了特定于区间的计算,则使用的先前值是存在时间(相对于最新值)等于请求者收集时间间隔的值。

例如,假设 CDP\_DP\_REFRESH\_INTERVAL 设置为 120 秒,attribute A 具有以下采样值:

时间	采样值
最新	2800

时间	采样值
2 分钟(120 秒)前	2600
4 分钟(240 秒)前	2499
6 分钟(360 秒)前	1500
8 分钟(480 秒)前	1200
10 分钟(600 秒)前	1000

如果未选择**特定于区间的计算**,那么 delta 函数始终返回 200,即两个最新值的差值,计算公式为 2800 - 2600。无论此值是显示在 Tivoli Enterprise Portal 或 IBM Cloud Application Performance Management 控制台中、用于情境还是历史数据收集,都返回相同值。

如果选择了特定于区间的计算,那么 delta 函数返回依赖于请求者收集时间间隔的值。

如果某派生属性与 delta 函数用于收集时间间隔为 4 分钟的情境,那么 delta 函数返回的值是 301,即最新值与 4 分钟前获取的值的差值,计算公式为 2800 - 2499。

如果某派生属性与 rate 函数用于收集时间间隔为 10 分钟(600 秒)的情境,那么 rate 函数返回的值是 3,即最新值与 10 分钟前获取的值的差值除以此时间间隔秒数,计算公式为(2800 - 1000) / 600。

**注:** Tivoli Enterprise Portal 没有固有的收集时间间隔,因此 Tivoli Enterprise Portal 的变化量和速率计算始终使用最新属性值和第二新属性值,无论是否选择了**特定于区间的计算**,结果都相同。

为使 delta 和 rate 可以与特定于区间的计算正确地配合使用,

·代理程序必须定期地在后台收集数据,而不是按需收集(CDP\_DP\_THREAD\_POOL\_SIZE 必须大于 0)。

- ·使用属性的每个情境或历史数据收集时间间隔必须是后台刷新时间间隔 (CDP\_DP\_REFRESH\_INTERVAL) 的倍数。
- ·count(last 函数的第二个自变量)的大小必须足以容纳情境或历史数据收集的最大的收集时间间隔。例如,如果代理程序必须支持 10 分钟(600 秒)的历史数据收集,并且 CDP\_DP\_REFRESH\_INTERVAL 是 120 秒,那么 count 必须至少为 6,计算公式为 1+(600 / 120)。count 值为 6 可确保 last 函数返回 最新的采样,并且多达过去 600 秒的采样。

注:如果不满足这些条件,那么输入值很可能无效,并返回结果 0。

### 编辑派生属性

使用 Formula Editor 编辑派生属性。

"**属性信息**"页面上针对派生属性提供了 Formula Editor,如<u>第 33 页的『创建派生属性』</u>所述。有关 Formula Editor 的更多信息,请参阅 <u>第 39 页的『Formula Editor』</u>

### 移除属性

您可以使用"数据源定义"页面移除一个或多个属性。

### 过程

· 要移除一个或多个属性,右键单击属性,然后从显示的菜单中选择**移除**。 注:无法移除由派生属性使用的属性。必须先移除派生属性对您要移除的属性的引用。

#### 用于定义特性的字段和选项

特性信息页面上适用于所有数据源的字段信息及选项的描述

有关每个数据源的特定字段信息的信息,请参阅每个数据源的相关文档。

表 5. 用于定义特性的字段和选项			
字段名称/选项	描述	可接受的值	
特性名称	特性的名称,此名称将显示在 Tivoli Enterprise Portal 或 IBM Cloud Application Performance Management 控制台中	包含以下字符的字符串: ·A-Z ·_ ·a-z ·0-9 注:名称必须以A-Z或a-z开头。 特性名限长 63 个字符,属性组名 限长 63 个字符	
帮助文本	特性的帮助文本	字符串	
隐藏 - 只能用于派生特性	如果选中此选项,那么此特性不会 显示在 Tivoli Enterprise Portal 或 IBM Cloud Application Performance Management 控制 台中。请参阅最后一行中的注释。	不适用	
从其他特性值派生	特性值将根据其他特性的值进行计 算	不适用	
键特性	特性是表中的键。请检查此特性是 否有助于唯一定义所报告的对象。 如果对数据进行仓储和摘要,那么 将使用键特性在摘要表中进行数据 汇总。	此选项不适用于 Perfmon 特性。	
特性信息窗格	此选项卡的内容取决于此特性所属的 关您要监视的数据源的信息,以获取 对于派生特性,请在 <b>公式</b> 字段中输力 量的特性值。您可以在 <b>公式</b> 字段中转 形公式编辑器。请参阅 <u>第 39 页的</u>	的数据源的类型。请参阅本章中有 取更多详细信息。 入公式,以计算基于其他特性或常 俞入公式,或者单击 <b>编辑</b> 以使用图 『Formula Editor』。	
特性类型	描述此特性在 Tivoli Enterprise Portal 或 IBM Cloud Application Performance Management 控制 台中的显示方式。共有 3 种类型: ·字符串 ·数字 ·时间戳记 <u>第 36 页的『属性类型』</u> 包含有 关特性类型的更多信息。	<u>第37页的表6</u> 包含数字特性类 型值的描述。	

表 5. 用于定义特性的字段和选项 (续)			
字段名称/选项	描述	可接受的值	
枚举	这可以是不带小数位的数字或字符 串值。	请通过使用 <u>第38页的『指定属</u> 性的枚举』中的过程,将您的枚举 项添加到表中。	
		在此特性中接收到来自代理程序的 对应值时,枚举名称将显示在 Tivoli Enterprise Portal 或 IBM Cloud Application Performance Management 控制台中。	
		此特性用于一组具有指定含义的特定值(例如,1代表 UP,2代表 DOWN)。	
<b>注:</b> 在计算中将此特性与其他特性配合使用时,有多个理由支持不显示基本值。例如,表示字节计数的数字回绕太快,因此毫无用处。			

### 属性类型

共有三种属性类型

这三种属性类型是:

- ·字符串
- ·数字
- ·时间戳记

### 字符串属性

当您选择字符串时,请使用最大大小字段来指定字符串的最大长度(字节)。缺省大小是 64 个字节。

字符串值可以包含任何 UTF-8 字符。最大大小是为了包含该字符串而分配的缓冲区的总长度(字节)。某些 非 ASCII UTF-8 字符占用多个字节,因此您选择最大大小时必须考虑到此部分空间。仓库中的数据聚集将显 示期间内收集到的最新值。

### 数字

当您指定数字时,可以设置多个选项。请参阅<u>第37页的表6</u>,以获取有关这些选项的信息。

### 时间戳记

时间戳记属性是遵循 CYYMMDDHHMMSSmmm 格式(其中 C 为 1 时表示 21 世纪)的字符串属性。对于脚本或 套接字客户机,这 16 个字符必须全部使用。在 Tivoli Enterprise Portal 或 IBM Cloud Application Performance Management 控制台中显示时,时间戳记属性类型将以语言环境的正确格式显示。

在使用 WMI 的浏览功能时, Agent Builder 会自动将 CIM 类型为 CIM\_DATETIME 的属性标记为时间戳记。 数据提供程序会自动将 WMI 属性转换成此格式。

### 属性的数字方面

对属性的大小、用途、精度和范围方面的描述。

指定数字属性时,必须指定属性的大小、用途、精度和范围。有关更多信息,请参阅第37页的表6。

表 6. 数字属性选项			
数字方面	选项和字段	描述	
大小	32位 64位	32 位数字值的范围可以从 - 2147483648 到 2147483647(大约 从 - 2,000,000,000 到 2,000,000,000)。	
		64 位数字值的范围可以从 -9223372036854775808 到 9223372036854775807(大约从 -9x10 <sup>18</sup> 到 9x10 <sup>18</sup> )	
用途	标尺	返回的原始值大于或小于先前值的整数值。支持负值。此类型为整 数的缺省类型。仓库中的数据聚集产生最小值、最大值和平均值。	
	计数器	这是正整数值,包含通常随时间推移而增加的原始值。仓库中的数据聚集显示增量值的总值、上限值、下限值以及最新值。在以下基于增量的计算示例中(一个小时内的详细数据值为9、15、12、20和22),基于增量的处理具有下列规则:	
		·如果当前值大于或等于先前值,那么输出等于先前值减去当前值	
		·如果当前值小于先前值,那么输出等于当前值	
		·由于 15 大于 9,因此输出等于 6	
		·由于 12 小于 15,因此输出等于 12	
		·由于 20 大于 12,因此输出等于 8	
		·由于 22 大于 20,因此输出等于 2	
		·TOT_值为28,这是输出总和	
		·LOW_值为 2,这是最小的输出	
		·HI_值为12,这是最大的输出	
	属性	这是不常变化的对象的属性。仓库中的数据聚集将显示在周期内收集到的最新值。	
	变化量	此整数值表示此属性的当前值和先前值的差值。由于此属性在仓库 中表示为标尺,因此仓库中的数据聚集将产生最小值、最大值和平 均值。	
	变化百分比	此整数值表示当前值和先前值的变化百分比。此类型的计算方法 为: ((new -old)*100)/old。由于此类型在仓库中表示为标 尺,因此仓库中的数据聚集将产生最小值、最大值和平均值。	
	变化率	这是一个整数值,表示的含义是当前值与先前值的差值除以采样间 隔秒数。它将"值(例如字节数)"转换为"值/秒(字节数/ 秒)"。由于此类型在仓库中表示为标尺,因此仓库中的数据聚集 将产生最小值、最大值和平均值。	
小数位	小数位调整	"小数位"确定数字中有多少小数位。每个小数位以因子 10 来缩 小之前提到的范围。例如,小数点调整为 2 时,将显示两个小数 位,而在 32 位数字中,允许的范围变成 - 21474836.48 到 21474836.47。	
		指定非零的小数点调整时,会在内部将数字处理为浮点数。因此, 较大的 64 位数字的精度可能会降低。	
范围	最小值 最大	"范围"给出值的预期范围。如果没有给出最小值或最大值范围, 那么将使用先前描述的最大值。此范围用于在某些图形 Tivoli Monitoring 工作空间视图中产生更有用的初始视图。	

### 指定属性的枚举

使用"属性信息"页面指定值枚举。

#### 关于此任务

为属性指定枚举是一个简短的过程。当遇到定义了枚举的值时, Tivoli Enterprise Portal 或 IBM Cloud Application Performance Management 控制台中会显示此枚举名称, 而不是值。

### 过程

- 1. 在"属性信息"页面的属性类型区域中,单击数字。
- 2. 在**枚举**区域中,单击枚举,然后单击**添加**。
  - 这将显示"**枚举定义**"窗口。
- 3. 在窗口的字段中输入枚举的名称和值。
- 4. 单击确定。

然后,您可以添加更多的枚举。

### 指定用作状态指示符的特性的严重性

在 IBM Cloud Application Performance Management 环境中,摘要仪表板必须显示状态。您必须使用特性来提供状态值。对于此特性,必须指定值,以表示特定的状态严重性。

#### 关于此任务

用于指示状态的特性必须是数字。请在"**仪表板设置**"向导中选择此特性;有关如何使用此向导的指示信息,请参阅<u>第 187 页的『第 12 章 为 Cloud APM 准备代理程序』</u>。

您可以为特性指定对应于"正常"、"警告"和"严重"严重性的值。任何其他值均表示"未知"严重性状态;您还可以将某些值显式定义为"未定义",对于这些值,用户界面会显示"未知"状态。

#### 过程

1. 选择要编辑的特性。

页面的"特性信息"窗格将更新,以显示所选特性的属性。

- 2. 在"特性信息"窗格中,单击严重性选项卡。
- 3. 选择必要的严重性("正常"、"警告"、"严重"和"未定义"),并单击编辑。
- 4. 选择**范围**或单个数字, 输入值范围或单个数字值, 然后单击确定。
- 5. 可选:如果需要为同一严重性添加其他值(例如,2和25都表示警告),请单击添加,选择严重性,输入值,然后单击确定。

### 过滤属性组

您可以创建过滤器,以限制从返回采样数据的属性组返回的数据。

### 开始之前

如果存在属性组,打开"数据源定义"页面。有关更多信息,请参阅<u>第27页的『"数据源定义"页面』</u>。 如果要创建属性组,请执行<u>第14页的『定义初始数据源』</u>中的步骤,然后单击初始数据源信息页面中的**高** 级。

### 过程

1. 使用下列其中一个步骤以开始创建过滤器:

- ·如果要创建属性组,单击初始数据源信息页面中的高级。
- ·如果存在属性组,请在"数据源定义"页面中选择该属性组,然后在"数据源定义"页面中单击高级。

2. 在"高级数据源属性"页面中,输入选择公式。您输入的选择公式的求值结果必须是布尔值 true 或 false。

在"高级数据源属性"页面中,您可以单击编辑,以使用公式编辑器来输入或修改公式。有关 Formula Editor 的更多信息,请参阅 <u>第 39 页的『Formula Editor』</u>

3. 在完成过滤器选择公式输入之后,单击确定直到返回到"数据源定义"页面。

创建过滤器后,代理程序会使用该过滤器对每行数据求值。当某一行数据的过滤器求值为 true 时,此数 据将发送到 IBM Tivoli Monitoring 或 IBM Cloud Application Performance Management。当过滤器求值 为 false 时,不发送此行数据,会将其废弃。

### 下一步做什么

您可以使用属性组的测试功能来验证过滤器是否按预期工作。有关属性组测试的更多信息,请参阅<u>第191页</u>的『属性组测试』

# **Formula Editor**

使用 Formula Editor 创建和更改 Agent Builder 中的公式。

Formula Editor 是一个图形工具,在执行以下任务时显示:

- 1. 创建或编辑派生属性,请参阅第 33 页的『创建派生属性』和第 34 页的『编辑派生属性』
- 2. 创建过滤属性组,请参阅第 155 页的『创建过滤属性组』
- 3. 从属性组过滤数据,请参阅第38页的『过滤属性组』



# 注意:

- · 在创建派生属性时,所创建的公式必须产生与此属性类型匹配的数据类型。例如,如果派生属性类型是数字,那么所创建的公式必须求值为数字结果。
- ·在创建过滤属性组或从属性组过滤数据时,所创建的公式必须产生布尔值"true"或"false"。

注: 在下列视图中,显示了 Formula Editor 正在为派生属性创建公式。在将 Formula Editor 与过滤属性组或 从属性组过滤数据结合使用时,这些视图是相同的。这些视图根据用途显示标题"派生 Formula Editor"或 "过滤器 Formula Editor"。

显示 Formula Editor 时,当前公式会装入编辑器。如果公式不存在,可以通过在 Formula Editor 窗口中的 公式空白处直接输入来输入公式。或者,可以单击插入以使用编辑器菜单选项开始输入公式。此编辑器在缺 省窗口中包含两个公式视图,以及第三个视图的选项:

#### 组件视图(缺省)

所编辑公式的组件显示在"操作数"区域和运算符字段中。可以使用选择菜单来编辑运算符以及它的两个操作数。

### 公式视图(缺省)

完整公式位于该窗口中的公式字段中。通过在此框中输入内容可以编辑公式。

#### 公式层次结构树视图(选项)

可通过选中**显示公式层次结构**复选框来显示公式层次结构树。在以后调用 Formula Editor 时会记住该复选框的状态。

### 更改 Formula Editor 组件视图

更改 Formula Editor 中的组件视图。

### 关于此任务

可通过以下方法更改组件视图中显示的组件:

### 过程

- · 将光标移到公式文本中。
- · 选择公式层次结构树中的其他节点。

· 选择上移一层或其中一个编辑按钮。

### 组件类型

您可以使用 Formula Editor 来编辑当前组件或者该组件的任何操作数或函数自变量。某些组件在 Formula Editor 中选中后可能会有不同的显示方式。

### Formula Editor Attribute 组件

使用 Formula Editor 中的 Attribute 组件可选择和操作公式中的属性。

### 关于此任务

您可以在 Formula Editor 的组件视图中,从属性列表选择属性组的属性。

### 过程

- **1**. 要处理特定属性,从列表中选择该属性,然后单击**编辑**。 这将显示"**编辑选定的属性**"窗口。
- 2. 可以通过以下方法操作所选属性:
  - · 您可以选择**字符串**或**数字**,将属性替换为字符串或数字。属性列表将替换为输入字段,且其内容将不 再与有效属性名列表进行比较。
  - · 您可以单击**函数**将属性替换为函数。这样会在名称后面添加圆括号,并且列表中现在将包含可供选择 的有效函数名。
  - · 可以输入属性名称而不是选择一个名称。如果尚未定义此属性组中的所有属性,那么输入名称将非常有用。
    - 如果不存在具有所输入名称的属性, 那么将显示警告。
    - 如果输入的字符不能用作属性名称的一部分,那么将显示错误。
    - 在更正警告或错误之前,确定按钮处于禁用状态。
  - · 不会基于类型来过滤属性。如果选择或输入了错误类型的属性(或任何值),将显示警告消息。

#### Formula Editor Literal 组件

使用 Formula Editor 中的字符串和数字组件来操作公式中的字面值。

### 关于此任务

字面值是在公式中直接输入的任何值,它不来自属性值或函数。字面值可以是字符串或数字。

### 过程

- ·您可以通过单击属性将文字串或数字替换为属性。在选择或输入有效的属性名称时不得带引号。
- · 您可以通过单击**函数**将文字串或数字替换为函数。这样会在名称后面添加圆括号,并且选择列表将包含可供选择的有效函数名。
  - 如果在需要字符串的地方输入了数字,那么将显示警告,反之亦然。
  - 如果选择了数字, 而字段内容不是数字, 那么将显示错误。在更正错误之前, 确定处于禁用状态。

#### Formula Editor Operator 组件

使用 Formula Editor 中的 Operator 组件来操作公式中的运算符。

### 关于此任务

Operator 组件显示运算符及其操作数。

### 过程

· 在 Formula Editor 组件视图中,从运算符列表中选择两个操作数之间的运算符。运算符 (%) 会将第一个操作数乘以 100,然后除以第二个操作数。

- ·选择运算符(+、-、\*、/ 或 %)。
  - 页面的左操作数部分位于运算符前面。
  - 右操作数部分位于运算符后面。
  - 如 <u>第 40 页的『Formula Editor Attribute 组件』</u>和 <u>第 40 页的『Formula Editor Literal 组件』</u>所述, 可以编辑简单的操作数(属性和字面值),而无需将所选组件更改为操作数。
  - 可通过单击编辑来编辑复杂操作数(由其他运算符或函数组成)。此操作会突出显示操作数组件而不 是整个运算符。

### Formula Editor 条件表达式组件

条件表达式组件显示一个条件、条件为 true 时返回的值以及条件为 false 时返回的值。

- "条件"部分中的表达式求值为 true 或 false。运算符 (==)、(!=)、(<)、(<=)、,(>)、(>=)、
   (&&)、(||)、(!)可用于构成返回 true 或 false 的表达式。
- ·如 <u>第 40 页的『Formula Editor Attribute 组件』</u>和 <u>第 40 页的『Formula Editor Literal 组件』</u>所述,可以 编辑简单的操作数(属性和字面值),而无需将所选组件更改为操作数。
- ·可通过单击 Edit 来编辑复杂操作数(由其他运算符或函数组成)。此操作会突出显示操作数组件而不是整个条件表达式。
- ·请参阅 <u>第 41 页的</u>『Formula Editor 通用选项』以了解有关使用以下选项的信息: 插入、移除、上移一层 和编辑。

### 相关概念

<u>第 39 页的『Formula Editor』</u> 使用 Formula Editor 创建和更改 Agent Builder 中的公式。

### Formula Editor Function 组件

使用 Formula Editor 中的 Function 组件可选择和操作公式中的函数组件。

### 关于此任务

Function 组件显示函数及其自变量。

### 过程

- · 要使用函数,可从 Formula Editor 的列表中选择函数名。
  - 选定函数的描述将显示在函数后。
  - **函数自变量**部分显示在函数名后面。将显示选定函数的相应自变量数量。将显示特定于选定函数的描述。
  - 如 第 40 页的『Formula Editor Attribute 组件』和 第 40 页的『Formula Editor Literal 组件』所述, 可以编辑简单的自变量(属性和字面值), 而无需将所选组件更改为操作数。
  - 可通过单击编辑来编辑复杂的自变量(由运算符或其他函数组成)。此操作会突出显示自变量组件而 不是整个函数。
- · 对于自变量个数不固定的函数,除了 第 41 页的『Formula Editor 通用选项』中描述的操作之外,还可以单击插入来添加自变量,或单击移除来移除自变量。
- · 对于 getenv 函数, 可通过单击插入来选择配置属性。如果选择"配置属性"选项, 则会显示"配置属性" 窗口。

### Formula Editor 通用选项

您可以在 Formula Editor 的所有视图中使用某些选项

Formula Editor 通用选项包括:

- ·插入
- ·卸下
- ·上移一层

·编辑

### 插入

**插入**在组件前面插入运算符或函数。该组件将降级为运算符的一个操作数或函数的一个自变量。例如,如果在 sqrt(attr2)函数前单击**插入**,系统将询问您要插入什么内容,并将显示以下选项:

·以 sqrt(attr2) 作为其中一个运算符操作数的运算符

·以 sqrt(attr2) 作为函数第一个自变量的函数

·以 sqrt(attr2) 作为 true 或 false 值的条件表达式

如果在 getenv 函数前单击插入,系统将询问您要插入什么内容,并将显示以下选项:

- ·**配置属性**:使用此选项来检索已为代理程序设置的配置属性的值,或者检索运行代理程序的主机上任何环境变量(例如, JAVA\_HOME)的值。
- ·以 attr2 作为其中一个运算符操作数的运算符
- ·以 attr2 作为函数第一个自变量的函数
- ·以 attr2 作为 true 或 false 值的条件表达式

### 移除

移除仅可用于运算符和函数,与插入效果相反。您单击移除时,系统将询问您用什么来替换移除的运算符或 函数。例如,sqrt(attr2)函数前的移除将显示以下选项:

·当前自变量1,attr2

·新字符串、数字或特性引用

选择新字符串、数字或特性引用将废弃待移除点之后的整个树,并将其替换为新特性或字面值。

单击**当前自变量**可提升所选操作数或自变量以替换移除的运算符或函数。如果有更多自变量或操作数,您可以单击其后的选项。将废弃任何其他操作数或自变量。

### 上移一层

单击上移一层将在树中上移。

#### 编辑

单击复杂操作数或自变量前的编辑可使其成为待编辑组件。

在单击编辑之后单击上移一层可将当前组件复原为单击编辑之前的状态。

### Formula Editor - 公式错误

更正 Formula Editor 中的公式错误

此组件视图在没有公式或无法解析输入的公式时有所不同。它不显示公式树。而是显示错误消息。

要更正含有解析错误的公式,可以直接在公式字段中进行输入,也可以单击**插入**将其替换为新公式。在此情况下,**插入**呈现以下选项:

- ·属性
- ·字符串
- ·数字
- ·运算符
- ·条件表达式
- ·函数
- 相关概念

第 39 页的『Formula Editor』

# 公式运算符和函数

可以在 Formula Editor 中使用的公式运算符和函数的参考(包括示例)。

派生的属性值是根据同一数据源中的常量和其他属性值对表达式求值的结果。表达式语法是常规数学表达式-操作数 运算符 操作数(以及用于分组的圆括号)。通过使用一般数学运算符 +、-、\*、/和 %来组合数 字属性和其他数字属性或常量,%用于将**左操作数**乘以100,然后再除以**右操作数**。可使用 + 将字符串属性 与其他字符串属性或常量组合。还可以使用以下描述的函数。使用以下格式输入函数: function\_name(argument\_1, argument\_2, argument\_3)。

属性通过其名称表示(与您在"**数据源**"信息树中看到的名称相同)。整数常量指定为数字。字符串常量用 引号括起。

可以在公式中使用以下函数:

abs

返回某个数字的绝对值

#### atof

将字符串转换为浮点值

#### atoi

将字符串转换为整数值。其运算方法与一般 C atoi 的工作原理相同:在第一个非十进制字符处停止。

#### average

返回单个值,该值是一组值中的平均值。这组值来自函数的自变量。可以提供多个单独的值(例如属性 名称或常量),每个值位于单独的自变量中。或者,也可将 last 函数作为此函数的唯一自变量(以计算 属性的最近值的平均值)。

此函数的使用示例包括:

average (Attr\_A, AttrB, Attr\_C)

average (last (Attr\_A, 10))

#### ceiling

返回不小于自变量的最小整数。

例如,如果 attribute\_a = 12.4,那么 ceiling(attribute\_a)返回值 13。如果 attribute\_a = -12.4,那么 ceiling(attribute\_a)返回值 -12。

#### delta

属性的最新值与先前针对该属性收集到的某个值的差。delta 的单一自变量必须是 last 函数,后者用于 获取属性的当前值和先前值。常规用法如下所示:

delta (last(OtherAttribute, 2))

有关使用 last 函数中的哪些属性值来计算 delta 的更多信息,请参阅<u>第 33 页的『特定于区间的计算』</u>。 此函数仅适用于派生属性,而不适用于属性组过滤器。

#### floor

返回不大于自变量的最大整数。

例如,如果 attribute\_a = 12.4,那么 floor(attribute\_a)返回值 12。如果 attribute\_a =-12.4,那么 floor(attribute\_a)返回值 -13。

#### getenv

返回提供的环境或"配置变量"的值。

#### ipAddressToName

将 IP 地址转换为主机名。此函数需要一个自变量(采用点分十进制表示法的 IP 地址字符串)。如果无法解析该地址,那么会返回 IP 地址。

itoa

将整数转换为字符串。当您想要将数字值连接成字符串时,此函数非常有用。派生字符串+函数仅采用 两个字符串自变量。

#### last

返回一列值以供min、max、average、stddev、rate 和 delta 函数使用。它采用两个自变量:要 收集的属性以及要在计算中使用的值的数量。如果所需的属性是"用字符串表示的整数值"属性,那么 第一个自变量可包含 atoi 函数,例如 atoi(numericalStringAttribute)。第二个自变量必须为 数字。它可以硬编码为常量,也可以是 atoi(getenv("ENV\_VAR"))表达式的结果。它不能引用任何 属性值。

此函数的使用示例包括:

average (last (Attr\_A, 10))

last (Attribute\_A, \${K01\_NUM\_COLLECTIONS}))

限制:在特定公式中,只能使用一次 last 函数。

#### matches

返回布尔值 true 或 false,以指示正则表达式是否与某个值匹配。此函数采用两个自变量,即字符串源 以及结果要与此字符串进行比较的正则表达式。此函数对于过滤属性组而言非常有用。

#### max

返回单个值,该值是一组值中的最大值。这组值来自函数的自变量。可以提供多个单独的值(例如属性 名称或常量),每个值位于单独的自变量中。或者,也可将 last 函数作为此函数的唯一自变量(以计算 属性的最近值的最大值)。

#### min

返回单个值,该值是一组值中的最小值。这组值来自函数的自变量。可以提供多个单独的值(例如属性 名称或常量),每个值位于单独的自变量中。或者,也可将 last 函数作为此函数的唯一自变量(以计算 属性的最近值的最小值)。

#### nameToIpAddress

将主机名转换为 IP 地址。此函数需要一个自变量,即主机名字符串。如果无法解析该地址,那么会返回 主机名。

#### **NetWareTimeToTivoliTimestamp**

将 Novell NetWare 十六进制时间值转换为 Tivoli Monitoring 时间戳记。此函数需要一个自变量,即一个 特殊的 NetWare 十六进制时间值。属性类型为时间戳记。

#### rate

属性的最新值与先前针对该属性收集到的某个值的变化率(每秒)。rate 的单一自变量必须是 last 函数,后者用于获取属性的当前值和先前值。常规用法如下所示:

rate (last(OtherAttribute, 2))

有关使用 last 函数中的哪些属性值来计算 rate 的更多信息,请参阅<u>第 33 页的『特定于区间的计算』</u>。 此函数仅适用于派生属性,而不适用于属性组过滤器。

#### replaceFirst

将匹配正则表达式的第一个子串替换为替换字符串。此函数采用三个自变量。第一个:输入字符串。第 二个:用于匹配输入字符串中的子串的正则表达式。第三个:替换字符串。有关正则表达式和替换字符 串中允许使用的替换值的详细信息,请参阅(第 309 页的『附录 F ICU 正则表达式』)。

#### replaceAll

将匹配正则表达式的所有子串替换为替换字符串。此函数采用三个自变量。第一个:输入字符串。第二 个:用于匹配输入字符串中的子串的正则表达式。第三个:替换字符串。有关正则表达式和替换字符串 中允许使用的替换值的详细信息,请参阅(<u>第 309 页的『附录 F ICU 正则表达式』</u>)。

#### round

将数字四舍五入至最近的整数。

#### sqrt

返回数字的平方根

#### stddev

返回单个值,该值是一组值的标准差。这组值来自函数的自变量。可以提供多个单独的值(例如属性名称或常量),每个值位于单独的自变量中。或者,也可将 last 函数作为此函数的唯一自变量(以计算属性的最近值的标准差)。

#### **StringToTivoliTimestamp**

将日期和时间字符串转换为 Tivoli Monitoring 时间戳记。此函数需要两个自变量。第一个自变量是以自由格式字符串表示的时间戳记。第二个自变量是识别如何解析以自由格式字符串表示的时间戳记的格式字符串。(<u>第 45 页的表 7</u>) 描述了有效格式参数。属性类型为时间戳记。

表 7. StringToTivoliTimestamp的有效格式参数				
符号	含义	格式	示例	
У	年份	уу	96	
		уууу	1996	
М	月份	M 或 MM	09	
	<b>注:</b> 仅支持以英语表示的 日份字符串。	ммм	Sept	
	1111111111	мммм	September	
d	日期	d	2	
		dd	02	
E	周历日	EE	Sa	
	<b>注:</b> 仅支持以英语表示的 周历日字符串。	EEE	Sat	
		EEEE	Saturday	
h	以 AM 或 PM 表示的钟 点 (1-12)	hh	07	
Н	一天中的钟点 (0-23)	нн	00	
m	小时中的分钟	mm	04	
S	分钟中的秒	SS	05	
S	毫秒	S	2	
		SS	24	
		SSS	245	
a	AM 或 PM 标记	a 或 aa	ат	
任何其他 ASCII 字符	跳过此字符	-(连字符)		
		(空格)		
		/(正斜杠)		
		:(冒号)		
		* (星号)		
		, (逗号)		

第46页的表8提供了时间戳记的字符串表示和用于解析它们的格式字符串。

表 8. StringToTivoliTimestamp 示例. 下表列出并说明了时间戳记字符串表示的一些示例				
时间戳记的字符串表示 格式字符串				
96.07.10 at 15:08:56 yy.MM.dd ** HH:mm:ss				
Wed, August 10, 2010 12:08 pm	EEE, MMMM dd, yyyy hh:mm a			
Thu 21/01/2010 14:10:33.17	EEE dd/MM/yyyy HH:mm:ss.SS			

#### sum

返回单个值,该值是一组值的总和。这组值来自函数的自变量。可以提供多个单独的值(例如特性名称 或常量),每个值位于单独的自变量中。或者,也可将 last 函数作为此函数的唯一自变量(以计算特性 的最近值的总和)。

#### **TivoliLogTimeToTivoliTimestamp**

将 Tivoli 日志文件时间戳记转换为 Tivoli Monitoring 时间戳记。此函数需要一个自变量,即来自 Tivoli 日 志文件的字符串时间戳记。特性类型为时间戳记。

#### tokenize

标记化的字符串的一个标记。此函数需要三个自变量。第一个自变量是要分割成标记的字符串。第二个 自变量给出该字符串中用于分隔各个标记的一个或多个字符。任何时候出现此自变量中的任意字符时, 都将用于识别和分隔第一个自变量中的标记。第三个自变量是要作为此函数结果返回的标记的索引。第 一个标记为 index 0,第二个标记为 index 1,以此类推。此自变量还可以是返回最后一个标记的字符串 LAST。

#### UTCtoGMT

将世界标准时间转换为 GMT Tivoli Monitoring 时间戳记。此函数需要一个自变量,即整数 time\_t 值。特性类型为时间戳记。

#### UTCtoLocalTime

将世界标准时间转换为本地 Tivoli Monitoring 时间戳记。此函数需要一个自变量,即整数 time\_t 值。特性类型为时间戳记。

以下函数不采用任何自变量并返回数字。

#### count

保留一个计数器,首次调用时,起始值为1,之后每调用一次,值递增1。如果在同时使用 last 的表达 式中使用该函数,它将与 last()存储的元素数目匹配,但此过程仅持续到 last()达到其最大值。此 时,每增加一个新值,last()就将删除最旧的值,从而保持值总数不变,而 count()的值会不断增 加。

#### cumulativeSum

返回由流量控制摘要事件表示的重复事件的自变量值总和。或者如果自变量是来自数据源的单个事件, 返回该自变量。它采用单一数字自变量。此函数仅适用于开启了事件过滤和摘要功能的事件属性组。

#### eventThreshold

返回为生成事件的属性组配置的阈值。这是一个数字,具有以下三个枚举值:

- ·SEND\_ALL (-3)
- ·SEND\_FIRST (-2)
- SEND\_NONE (-1)

括号中的数字是原始值。但是, Agent Builder 会定义枚举,这样在缺省情况下,文本版本在 Tivoli Enterprise Portal 或 IBM Cloud Application Performance Management 控制台中可见。如果您指定了一个实际的数字阈值,且并非上述三个预定义选项之一,那么此函数将返回该数字。该值是大于 0 的整数。此函数仅适用于开启了事件过滤和摘要功能的事件属性组。

#### isSummaryEvent

如果是来自数据源的单个事件,那么返回 0,如果事件是流量控制摘要事件,那么返回 1。如果对该函数 使用缺省特性,那么显示的值是 Event 和 Summary Event。如果手动创建特性,那么除非您将名称定义 为枚举值,否则显示的值是 0 和 1。此函数仅适用于开启了事件过滤和摘要功能的事件属性组。

### occurrenceCount

返回由流量控制摘要事件表示的匹配事件数,如果是来自数据源的单个事件,那么返回1。(流量控制 摘要事件包括第一个事件)。此函数仅适用于开启了事件过滤和摘要功能的事件属性组。

#### summaryInterval

返回为生成事件的属性组配置的摘要时间间隔(以秒为单位)。此函数仅适用于开启了事件过滤和摘要 功能的事件属性组。

#### 示例

使用公式运算符和函数创建派生特性和过滤特性的示例

#### 示例 1 - 派生特性

如果您拥有定义了以下特性类型的数据源:

名称	字符串
xBytes	数字
yBytes	数字
Virtual_Size	数字

您可以定义:

·用于表示 xbytes 和 ybytes 之和的特性 totalbytes。输入公式 xBytes + yBytes。

 ·表示 yBytes 占字节总数百分比的特性 yPercent,可以定义为 yBytes % (xBytes + yBytes)或 yBytes % totalBytes。

### 示例 2 - 派生特性

以下公式返回 Virtual\_Size 特性的最近收集值中的最大值。收集的样本数是配置变量 *K4P\_COLLECTIONS\_PER\_HISTORY\_INTERVAL*(通过 getenv 访问)的值,该值将(通过 atoi)转换为数 字:

max(last(Virtual\_Size,atoi(getenv("K4P\_COLLECTIONS\_PER\_HISTORY\_INTERVAL"))))

### 示例 3 - 派生特性

以下公式返回 xbytes 和 ybytes 特性值的平方总和的平方根:

sqrt(xBytes \* xBytes + yBtyes \* yBytes)

### 示例 4 - 派生特性

以下公式返回属性组的最近 20 个样本中 xBytes 特性的平均值。如果自代理程序启动以来收集的样本数少 于 20、那么它将返回所有样本中 xBytes 特性的平均值:

```
average(last(xBytes,20))
```

#### 示例 5 - 过滤特性

您的数据源返回下列数据:

Name	Туре	Size	Used	Free
Memory	MEM	8	4	4
Disk1	DISK	300	200	100
Disk2	DISK	500	100	400

您只对磁盘使用情况感兴趣。解决方案是创建过滤器以限制返回的数据。要限制返回的数据,您可以创建返回布尔值 true 或 false 的简单过滤器,如下所示

磁盘过滤器:

Type=="DISK"

现在,当过滤器 Type=="DISK"为 true 时,此属性组仅返回磁盘使用情况数据,例如:

Name	Туре	Size	Used	Free
Disk1	DISK	300	200	100
Disk2	DISK	500	100	400

### 示例 6-过滤属性

您的数据源返回下列数据:

Name	Size	Used	Free
Memory	8	4	4
Disk1	300	200	100
Disk2	500	100	400

返回的数据与上一示例相似,但这次没有 Type 属性。在这里,您可以使用 matches 函数来查找名称属性值 与 "Disk" 后跟一个数字匹配的任何数据行。

磁盘过滤器:

matches(Name, "Disk[0-9]\*")

现在, 仅当此过滤器在属性 Name 中与字符串 "Disk" 后跟一个数字匹配时, 才会返回磁盘使用情况数据 行:

Name	Size	Used	Free
Disk1	300	200	100
Disk2	500	100	400

# 指定操作系统

当您定义并非在代理程序所支持的所有操作系统上均可用的数据源时,必须指定运行数据源所在的操作系统。

### 关于此任务

缺省情况下,数据源将在代理程序级别定义的所有操作系统上提供数据,如<u>第18页的『缺省操作系统』</u>所述。您可以为每个数据源更改操作系统。

#### 过程

1. 要打开"操作系统"部分,添加数据源时,请在"数据源信息"页面中单击操作系统。

2. 选择运行该数据源所在的操作系统。

请选择单个操作系统、所有操作系统、特定类型的所有操作系统或者代理程序缺省操作系统。

### 配置和调整数据收集

在创建 Agent Builder 代理程序后,您可以配置和调整其数据收集以实现最佳结果。

对于不同的 Agent Builder 代理程序乃至一个代理程序中的不同属性组,代理程序的配置和调整方法可能各不相同。Agent Builder 代理程序可以包含两种数据,而且它们对最常见的数据类型支持两种基本数据收集方法。

### 数据类型

代理程序收集两种数据:

- 1. 大多数 Tivoli Monitoring 属性组表示数据的快照。有些属性组会请求并返回数据。代理程序使用此类数据 来表示配置、性能、状态和其他信息,对于此类情况,一次收集一组数据较为合理。此数据称为采样数 据。
- 2. 有些 Tivoli Monitoring 数据表示事件。在这种情况下,当事件发生时,代理程序必须将数据转发到 Tivoli Monitoring。事件示例包括 SNMP 陷阱、Windows 事件日志条目以及写入日志文件的新记录。为简单起见,这些类型的数据将进行分组并称为事件数据。

#### 采样数据

需要采样数据时,将向代理程序发送针对特定属性组的请求。可通过单击 Tivoli Enterprise Portal 中的工作 空间发出请求。可发起请求的其他事件包括正在运行的情境、Warehouse 的数据收集或 SOAP 请求。代理程 序接收到该请求时,它将返回该属性组的最新数据。Tivoli Enterprise Portal 请求通过特定的受管系统名称 (MSN)定位特定属性组。情境和历史请求则更有意思,尤其是在代理程序包含子节点时。当某个情境需要子 节点中某个属性组的数据时,代理程序将接收到一个含有目标子节点列表的请求。代理程序必须先以所有子 节点的请求目标属性组的所有数据进行响应,然后 Tivoli Monitoring 才能处理下一个请求。

代理程序满足请求的最直接方式是,在每次接收到来自 Tivoli Monitoring 的请求时收集数据。Agent Builder 代理程序并非每次都收集数据。这是因为收集数据通常需要花费时间或耗用资源。而许多情况下会在短时间 内多次请求相同数据。例如,用户可能定义了多个按相同时间间隔对属性组运行的情境,而这些情境可能针 对多种不同的情况发出信号。其中每一个情境都会产生一条发送给代理程序的请求,但您可能希望所有情境 都得到相同的数据。因此,正因为每个情景都得到相同的数据,所以获取了更一致的结果,从而尽可能地降 低监视代理程序对系统资源的需求。

代理程序开发者可以对代理程序进行配置,通过选择下列其中一种方式运行收集来优化数据收集:

1. 随需应变收集:代理程序在收到请求时收集数据并返回该数据。

2. 预定收集:代理程序在后台按调度的时间间隔运行数据收集,并在收到请求时返回收集的最新数据。

在这两种方式下,代理程序都使用短期高速缓存。如果在高速缓存有效时收到另一条数据请求,代理程序将 返回高速缓存中的数据,而不会为每条请求收集新数据。使用高速缓存中的数据解决多个并发情景(和其他 类型)请求所导致的问题。环境变量定义数据的有效期、调度的收集时间间隔、用于收集的线程数以及代理 程序的运行方式(按需还是按调度运行)。您可以使用环境变量来调整每个代理程序,以使其在自己的环境 中实现最佳运行状态。

请参阅以下示例,这些示例说明了代理程序在这两种方式下的工作方式:

- ·代理程序1(按需收集):这是收集少量数据的一个简单代理程序,通常仅由情景或在 Tivoli Enterprise Portal 中偶尔访问。数据收集速度相当快,但可能会耗尽计算资源和网络资源。此代理程序通常定义为按 需运行。如果没有任何情境处于运行状态,或者没有任何用户在 Tivoli Enterprise Portal 中单击,那么代 理程序不执行任何操作。当需要数据时,将收集并返回数据。数据放置在短期高速缓存中,因此几乎同时 的未来请求将返回相同数据。这种收集对此代理程序而言很可能是最高效的运行方式,这是因为,它仅在 某人确实需要数据时收集数据。
- ·代理程序2(调度收集):这是一个复杂的代理程序,它包含子节点并从受监视资源的多个副本收集数据。资源的多个副本可以由一个代理程序管理。为了监视受监视资源的状态和性能,相对频繁地对数据运行情境是很常见的。此代理程序定义为运行调度收集。运行调度收集的其中一个原因是Tivoli Monitoring 代理程序评估情境的方式。由于情境在子节点中的属性组上运行,因此代理程序将同时从所有子节点接收到一个数据请求。在返回某个情境的所有数据之前,代理程序无法响应其他请求。如果代理程序在请求到达时收集了所有数据,那么当您在Tivoli Enterprise Portal 中单击代理程序的一个工作空间时,代理程序将冻结。为了避免冻结代理程序,Agent Builder 自动将所有子节点代理程序定义为在调度收集方式下运行。代理程序开发者将调整线程数和刷新时间间隔,以按合理的时间间隔收集此数据类型的数据。例如,刷新时间间隔可以是一分钟一次,也可以是五分钟一次。

### 环境变量

代理程序根据一组环境变量的值来确定使用哪种方式和如何运行调度数据收集。可以在"环境变量"面板上的代理程序定义中设置这些环境变量。每个环境变量都在菜单中与缺省值一起列出。也可以通过编辑 Windows 上的环境变量 (env) 文件或者 UNIX 上的初始化 (ini) 文件,针对安装的代理程序设置或修改环境 变量。控制采样属性组的数据收集的环境变量有: ·CDP\_DP\_CACHE\_TTL=<高速缓存数据的有效期 - 缺省值为 55 秒>

·CDP\_DP\_THREAD\_POOL\_SIZE=<用于并行收集的线程数 - 对于子节点代理程序,缺省值为 15>

- ·CDP\_DP\_REFRESH\_INTERVAL=<相邻两次收集间隔的秒数 对于子节点代理程序,缺省值为 60 秒>
- ·CDP\_DP\_IMPATIENT\_COLLECTOR\_TIMEOUT=<有效期到期后,等待新数据的时间长度 缺省值为 5 秒>

这些变量中,最重要的是 CDP\_DP\_CACHE\_TTL、CDP\_DP\_REFRESH\_INTERVAL 和 CDP\_DP\_THREAD\_POOL\_SIZE。

如果 CDP\_DP\_THREAD\_POOL\_SIZE 的值大于或等于 1,或者代理程序包含子节点,代理程序会在调度收集 方式下运行。如果未设置 CDP\_DP\_THREAD\_POOL\_SIZE 或者设置为 0,那么代理程序会在按需收集方式下 运行。

如果代理程序在调度方式下运行,那么代理程序每隔 CDP\_DP\_REFRESH\_INTERVAL 秒就自动收集所有属性 组。它使用一组后台线程执行收集。线程数通过 CDP\_DP\_THREAD\_POOL\_SIZE 设置。 CDP\_DP\_THREAD\_POOL\_SIZE 的正确值根据代理程序进行的操作而有所不同。例如:

- ·如果代理程序使用 SNMP 从远程系统收集数据,那么 CDP\_DP\_THREAD\_POOL\_SIZE 的值最好与受监视的 远程系统的数量相仿。通过将池大小设置为近似于受监视的远程系统的数量,代理程序将并行收集数据, 但是会限制远程系统上的并行负载。SNMP 守护程序往往会在繁忙时丢弃请求。丢弃请求将迫使代理程序 进入重试方式,从而导致耗用更多时间和更多资源来收集数据。
- ·如果代理程序包含许多需要花费较长时间来收集的属性组,那么请使用足够多的线程以使费时较长的数据 收集可以并行运行。对于其余属性组,您可以再添加一些线程。如果目标资源有足够的处理能力,请按此 方式使用线程。如果脚本运行很长时间或者 JDBC 查询花费很长时间,那么在这些情况下将花费较长时间 来收集属性组。

用较大的线程池运行代理程序将导致代理程序使用更多内存(主要用于为每个线程分配的堆栈)。但是这不 会显著增加进程的处理器使用率,也不会显著增加进程的实际工作集大小。使用适合于工作负载的正确线程 池大小可以使代理程序更为高效。您可以适当调整线程池大小,以使特定环境中的特定代理程序的行为符合 需要。

在收集数据时,数据存放在内部高速缓存中。此高速缓存用于满足收集新数据之前的未来请求。高速缓存的 有效期由 CDP\_DP\_CACHE\_TTL 控制。缺省情况下,有效期设置为 55 秒。当代理程序以调度方式运行时, 最好将有效期值设置为与 CDP\_DP\_REFRESH\_INTERVAL 相同的值。如果数据收集可能会持续很长时间,请 将该值设置为较大的值。使用此方式设置有效期时,在下次调度收集之前数据被视为有效。

最后一个变量是 CDP\_DP\_IMPATIENT\_COLLECTOR\_TIMEOUT。此变量仅当 CDP\_DP\_CACHE\_TTL 在收集 新数据前到期时才发挥作用。如果高速缓存在收集新数据前到期,代理程序会立即调度另一次数据收集。然 后它会等待此收集完成,但最多等待 CDP\_DP\_IMPATIENT\_COLLECTOR\_TIMEOUT 秒。如果该新收集操作 完成,就更新高速缓存并返回最新数据。如果该新收集操作未完成,就返回现有数据。在

CDP\_DP\_CACHE\_TTL 完成时代理程序不会清除高速缓存,从而避免发生 Universal Agent 所存在的问题。 Universal Agent 总是会在有效期结束时清除其数据高速缓存。如果 Universal Agent 在下次收集完成前清除 其数据高速缓存,该属性组的高速缓便是空的,在收集完成前不会返回数据。在情境运行时不返回数据就会 造成问题。任何在高速缓存清除后且下次收集未完成时运行的情境都将得不到数据,任何触发的情境都会被 清除。结果就是仅仅因为数据收集慢了一点,就有大量事件发生触发和清除。Agent Builder 代理程序不会引 发此问题。如果"旧"数据导致某情境触发,通常同样的数据会使该情境保持同样状态。在下次收集完成 后,该情境会获得新数据,并根据有效数据触发或清除。

### 属性组

Agent Builder 代理程序包含两个属性组,您可以使用这两个属性组来检查数据收集操作以及根据您的环境调整代理程序。这两个属性组是 Performance Object Status 和 Thread Pool Status。当使用这些属性组调整数据收集性能时,最有用的数据是:

·Performance Object Status 的 Average Collection Duration 属性。此属性向您显示每个属 性组收集数据所用的时间。通常,代理程序中较小百分比的属性组表示代理程序使用的大部分处理器使用 率或时间。您也许能够优化一个或多个此类属性组的收集。或者,如果某些数据不需要像其他数据一样保 持最新,那么可以修改一个或多个组的收集时间间隔。关于更多信息,请参阅(<u>第 51 页的『示例和高级</u> 调整』)。

- ·Performance Object Status 的 Intervals Skipped 属性。此属性显示代理程序尝试为属性组调 度新收集,而发现先前的收集仍在排队、等待运行或已在运行的次数。在行为正常的代理程序中,此属性 值对于所有属性组都是零。如果此数字开始增长,表示您应该调整数据收集,例如通过增加线程、延长收 集的时间间隔或优化收集。
- ·Thread Pool Status 的 Thread Pool Avg Active Threads 属性。您可以将此值与 Thread Pool Size 属性组比较,以了解对线程池的利用情况。如果平均活动线程数是 5,分配大小为 100 个线程的线程 池可能只会浪费内存。
- ·Thread Pool Status 的 Thread Pool Avg Job wait 和 Thread Pool Avg Queue Length 属性。这两个属性分别表示数据收集在队列中等待线程处理的平均时间和队列中的平均收集数。由于此数 据的这种收集方式,即使空闲时,系统也指示至少平均有一个作业在队列中等待。如果等待的作业数更多 或平均等待时间很长,表示收集需求未得到满足。您可以考虑增加线程,延长收集的时间间隔,或者优化 一个或多个属性组的收集。

#### 事件数据

Agent Builder 代理程序可以公开多种类型的事件数据。有些行为对于所有事件数据是相同的。代理程序将每个新事件作为单独的一行数据来接收。接收到事件数据行时,会立即将其发送到 Tivoli Monitoring 进行处理,并将其添加到代理程序中的内部高速缓存。将每一行发送到 Tivoli Monitoring 时,Tivoli Monitoring 都会执行情境和历史数据收集。高速缓存用于满足 Tivoli Enterprise Portal 或 SOAP 的数据请求。代理程序可以使用高速缓存来执行为属性组定义的重复检测、过滤和摘要。每个属性组的事件高速缓存的大小都由CDP\_PURE\_EVENT\_CACHE\_SIZE 设置。此高速缓存包含最近的 CDP\_PURE\_EVENT\_CACHE\_SIZE 事件,其中最新返回的事件排在第一位。每个事件属性组都有独立的高速缓存。当某个属性组的高速缓存写满时,就会从列表中删除最早的事件。

Agent Builder 代理程序可以公开下列各项的事件:

- ·Windows 事件日志条目
- ·SNMP 陷阱或通知
- ·添加到日志文件的记录
- ·JMX MBean 通知
- ·JMX 监视器
- ·来自 Java API 提供者或套接字提供者的事件。
- ·连接式属性组(其中一个数据源是事件数据源)

这些事件将按最适合每个源的方式处理。对于 SNMP 陷阱和通知、JMX 通知以及来自 Java API 和套接字提供者的事件,将以异步方式接收并立即转发到 Tivoli Monitoring。不需要调整这些收集器。代理程序使用Windows 事件日志 API 进行预订,以从操作系统接收 Windows 事件日志条目。如果代理程序使用较旧的事件日志记录 API,它会使用线程池设置轮询系统以接收新事件。对于数据源之一是事件数据源的连接式属性组,没有适用于该连接式属性组的调整。但任何适用于事件源组的调整都可使连接式属性组受益。

文件监视则更复杂。代理程序必须监视文件是否存在以及新记录添加到文件的时间。代理程序可以配置为使用文件名或静态名称的模式监视文件。由于符合模式的文件集合可能随着时间变化,因此代理程序每隔KUMP\_DP\_FILE\_SWITCH\_CHECK\_INTERVAL秒就会检查是否有新文件或更改的文件。此全局环境变量控制代理程序实例中的所有文件监视。当代理程序确定要监视的相应文件时,必须确定文件更改时间。在Windows系统上,代理程序使用操作系统 API 来侦听这些更改。代理程序会在文件更新时得到通知,并立即处理这些文件。在 UNIX 系统上,代理程序每隔 KUMP\_DP\_EVENT 秒检查一次文件更改。此全局环境变量控制代理程序实例中的所有文件监视。当代理程序注意到文件更改时,其处理文件中的所有新数据,然后等待下一次更改。

#### 示例和高级调整

#### 示例

在代理程序级别定义用于更高级调整的环境变量。您只要设置一次下列变量,它们就会应用于代理程序中的 所有属性组:

· CDP\_DP\_CACHE\_TTL

- ·CDP\_DP\_IMPATIENT\_COLLECTOR\_TIMEOUT
- ・KUMP\_DP\_FILE\_SWITCH\_CHECK\_INTERVAL
- $\cdot$  KUMP\_DP\_EVENT

您可以将下列变量应用于个别属性组。它们仍有适用于代理程序中所有其他属性组的全局设置:

- ·CDP\_DP\_REFRESH\_INTERVAL
- ·CDP\_PURE\_EVENT\_CACHE\_SIZE

如果您定义了包含下列六个属性组的代理程序:

- · EventDataOne
- $\cdot$  EventDataTwo
- EventDataThree
- SampledDataOne
- · SampledDataTwo
- $\cdot \, {\tt SampledDataThree}$

您可设置下列缺省变量:

- ·CDP\_DP\_CACHE\_TTL=55
- ・CDP\_DP\_IMPATIENT\_COLLECTOR\_TIMEOUT=2
- ·CDP\_DP\_REFRESH\_INTERVAL=60
- · CDP\_PURE\_EVENT\_CACHE\_SIZE=100

因此,对于所有包含采样数据的属性组(SampledDataOne、SampledDataTwo和SampledDataThree),都是每 60 秒收集一次。每个事件属性组(EventDataOne、EventDataTwo和EventDataThree)将在其高速缓存中存储最近的 100 个事件。

这些设置可能效果非常理想,也可能因为一些原因而必须在更详细级别控制这些设置。例如,如果 EventDataOne 通常接收到的事件是 EventDataTwo 和 EventDataThree 的 10 倍,该怎么办?再复杂一点, EventDataOne 和 EventDataTwo 之间其实是有关联的。在针对 EventDataTwo 收到一个事件时,对于 EventDataOne 总是存在多个事件,并且用户想要关联这些事件。因此,没有一个适合所有属性组的高速缓 存大小。最好让 EventDataOne 存储较多事件,让 EventDataTwo 存储较少事件。您可以通过将 CDP\_PURE\_EVENT\_CACHE\_SIZE 设置为适合大多数事件属性组的大小(100 似乎很合适)来实现这种存储 方案。然后,您可以将 CDP\_EVENTDATAONE\_PURE\_EVENT\_CACHE\_SIZE 设置为 1000。这样一来,所有 对应事件在 Tivoli Enterprise Portal 中都可见。

对 CDP\_DP\_REFRESH\_INTERVAL 可以进行同样处理。设置适合代理程序中大多数属性组的缺省值。然后, 针对必须单独收集的属性组设置 CDP\_attributegroup name\_REFRESH\_INTERVAL。为了优化收集,将 缺省 CDP\_DP\_REFRESH\_INTERVAL 设置为与 CDP\_DP\_CACHE\_TTL 值匹配。CDP\_DP\_CACHE\_TTL 是全局 值,因此,如果将其设置为小于刷新时间间隔的值,那么可能会发生意外的收集。

# 第6章定义和测试数据源

Agent Builder 支持多种数据提供程序。您可以从每种数据提供程序创建数据源。每种数据提供程序的数据源 创建与测试过程都不相同。

对于大部分数据提供程序,当您创建数据源时,会向代理程序添加数据集(属性组)。该数据集包含此数据 源收集的信息。

使用进程、Windows 服务或程序返回码数据提供程序的数据源使用特殊的"可用性"数据集。代理程序中只能创建一个"可用性"数据集。它包含此代理程序中,使用进程、Windows 服务或程序返回码数据提供程序的所有数据源收集的信息。

代理程序和子节点中的所有 Windows 日志数据源都将事件信息放置在一个"事件日志"数据集中。

#### 为 Cloud APM 设置数据源

在 Cloud APM 中,您可以使用来自于"详细信息"仪表板的所有数据集中的数据,并使用阈值管理器来设置 阈值。如果要在代理程序或子节点的摘要仪表板中使用来自数据集的信息(包括状态指示符)以及资源信息 (服务名称、地址和端口),那么数据集必须只能产生一行。

对于大部分数据提供程序,可以在数据集配置中选择**生成单一数据行**。如果收集的信息将包含多行,您可单击高级,以设置一个可确保生成正确行的过滤器(有关指示信息,请参阅<u>第38页的『过滤属性组』</u>)。您可以测试数据源以确保收集的信息产生所需的行。

对于某些数据提供程序,数据集必须产生多行。此外,进程、Windows 服务和命令返回码数据源会将数据放入产生多行的单个"可用性"数据集中。在此类情况下,必须创建产生单行的已过滤数据集。有关创建过滤型数据集(属性组)的指示信息,请参阅<u>第155页的『创建过滤属性组』</u>。

其他的一些数据提供程序产生事件数据;针对每个新事件包含一行。请勿将这些数据提供程序用于 Cloud APM 中的摘要或资源信息。

以下数据提供程序必须产生包含多行的数据集:

- ·进程(使用"可用性"数据集)
- ·Windows 服务(使用"可用性"数据集)
- ·程序返回码(使用"可用性"数据集)
- ·SNMP和 JMX(针对某些数据类型)
- ·套接字和 Java API(取决于应用程序)

以下数据提供程序产生事件数据:

- ·SNMP 事件
- ·日志文件
- ·AIX 二进制日志
- ·Windows 事件日志
- ·套接字和 Java API(取决于应用程序)

数据集的其中一个属性必须提供状态值。Cloud APM 将此值用作整体状态指示符。如果该行不包含可用作状态指示符的属性,那么可以创建派生属性以计算状态。您必须配置状态严重性值;有关指示信息,请参阅<u>第</u>38页的『指定用作状态指示符的特性的严重性』。

# 监视进程

您可以定义用于监视服务器上运行的一个或多个进程的数据源。这些进程与代理程序运行在同一主机上。对 于每个进程,数据源都会向"可用性"数据集添加一行。

过程

- 1. 在"代理程序初始数据源"页面或"数据源位置"页面中,单击监视数据类别区域中的进程。
- 2. 在**数据源**区域中,单击**进程**。
- 3. 单击下一步。
- 4. 在"进程监视器"页面上的"进程信息"区域中,提供显示名称和进程名称。您可以手动输入进程名称,也可以通过单击浏览来获取进程名称。单击浏览将显示本地系统或远程系统上当前正在运行的进程列表。

您可以通过选择**使用自变量匹配和匹配完整命令行**选项来进一步区分进程。例如,如果系统上正在运行 同一进程的多个实例,那么可以使用这些选项来区分这些实例。

表 9. "进程监视器"页面上的字段. 下表列出了"进程监视"页面中的字段及其描述				
字段名称	描述	可接受的值		
Display name	这是由进程所实现的应用程序组 件的描述性名称,显示在 Tivoli Enterprise Portal 或 IBM Cloud Application Performance Management 控制台中	描述性字符串		
Process name	正在受监视的进程的名称	有效的可执行文件名		
Use argument match	如果您想要对进程自变量进行匹 配,请选择此选项。	On 或 Off		
自变量	要对其进行匹配的自变量字符 串。自变量匹配查找所提供的作 为自变量子串的字符串。如果提 供自变量的任意部分作为输入字 符串,那么匹配成功。	字符串		
Match full command line	指定可执行文件的完整名称(可 能包含路径)	On 或 Off		
Command line	将所提供的字符串与用来启动进 程的标准命令名进行匹配。不包 含命令自变量。"标准"表示必 须包含该命令的路径。	字符串		
Operating systems	选择运行该进程的操作系统	任意选择		

5. 如果您单击**浏览**,则会打开"**进程浏览器**"窗口。该窗口初始包含有关 Agent Builder 系统上的每个进程 的详细信息。该信息包括标识、进程名称和进程的完整命令行。选择一个或多个进程,或者通过以下一 个或多个操作来使用"**进程浏览器**"窗口中的列表:

- a) 要对进程列表进行排序,请单击列标题。
- b) 要刷新窗口中的信息, 单击刷新(闪电)图标。
- c) 要搜索特定进程,单击**搜索**(望远镜)图标。 您可输入搜索短语并选择选项部分,以便按进程标识、名称和命令行执行搜索。
- d) 要查看其他系统上的进程,请从连接名称列表中选择先前定义的系统。或者,单击添加以输入新系统的系统信息。
  有关更多信息,请参阅第 56 页的『定义用于进程浏览的连接』。您可以同时从多个系统装入进程,

并且在为一个或多个连接装入进程时切换连接。

注: 在浏览远程系统时,仅当通过 Tivoli Enterprise Portal Server 浏览时,命令行详细信息才可用。 在以下示例中,选择 svchost.exe 之后,该进程将显示在进程监视器页面上的进程名称字段中(<u>第</u>55页的图1)。

🐵 IBM Tivoli Monitoring Agent Wizard 📃 🗖 🔀				
Process Monitor				
Enter the detail	s for the pr	ocess monitor.		
Process inform	ation			
Display name	svchost			
Process name	svchost.e	exe		Browse
-Matching				
Use argume	ent match			
Argument	:			Insert Property
Match full c	ommand lin	e		
Command line	;			Insert Property
- Operation Su	atoma			
AIX (32-bit)	stems	Linux 2.4 (Intel)	✓ Linux (64-bit Itanium)	✓ Windows
AIX (64-bit)		✓ Linux 2.6 (Intel)	✓ Linux (64-bit x86)	Windows (64-bit)
HP-UX (32-bit	t)	✓ Linux (31-bit zSeries)	Solaris (32-bit SPARC)	
HP-UX (64-bit	t)	Linux (64-bit zSeries)	Solaris (64-bit SPARC)	
HP-UX (64-bit	t Itanium)	Linux (64-bit PowerPC)	Solaris (64-bit x86)	
All operating	systems			
Agent derau	·			
0		< E	Back Next >	Finish Cancel

### 图 1. "进程监视器"页面示例

6. 使用第 54 页的表 9 中的信息填写"进程监视器"页面。

**注:** 如果在此监视器中描述的进程仅适用于您的应用程序运行所在的部分操作系统,那么可能需要使用相同显示名称来创建一个或多个进程监视器以涵盖其他操作系统。一次添加一个进程监视器。请确保各个监视器的显示名称相同,但是可以在选择的操作系统上找到该进程名称。

### 7. 执行下列其中一个步骤:

- ·如果您正在使用"代理程序"向导,请单击下一步。
- · 单击完成,以保存该数据源并打开 Agent Editor。

### 下一步做什么

如果想要在 IBM Cloud Application Performance Management 的摘要仪表板中使用来自此数据源的数据, 必须创建基于"可用性"数据集过滤的数据集(属性组),并将其配置为提供单行。使用"名称"字段来选择用于您的进程的行。

您可以使用"状态"字段来显示状态; DOWN 表示进程未在运行, UP 表示正在运行。在新的过滤属性组中, 选择"状态"字段并为其指定严重性值。

如果有多个进程副本正在运行, "可用性"数据集中将显示具有该进程名称的若干行, 且所有这些行都包含 UP 状态。您的过滤数据集必须配置为返回一行, 这样可以返回其中任何行, 但"状态"值在任何情况下均 有效。

有关指示信息,请参阅:

- ·第155页的『创建过滤属性组』
- · 第 38 页的『指定用作状态指示符的特性的严重性』
- ·第 187 页的『第 12 章 为 Cloud APM 准备代理程序』

### 定义用于进程浏览的连接

在定义进程数据源时,可以查看和选择来自其他系统的进程。但是,代理程序运行时,将监视与代理程序在 同一系统上运行的进程。

#### 关于此任务

您必须具有其他系统的凭证,否则这些系统必须由 Tivoli Monitoring 操作系统代理程序监视。

#### 过程

1. 要定义连接,请单击"**进程浏览器**"窗口中的**添加**。

您可以选择连接类型(Secure Shell (SSH)、Windows 或 Tivoli Enterprise Portal Server Managed System),或者选择现有连接以用作模板。

要添加 Managed System 连接,需要 Tivoli Enterprise Server 主机名、Tivoli Monitoring 用户名和密码。 您还需要远程连接的受管系统名称。选定受管系统后,该表将列出远程系统上的进程。

**注:** 操作系统代理程序必须在您正尝试浏览的系统上运行。代理程序还必须连接到正在运行的 Tivoli Enterprise Monitoring Server 和 Tivoli Enterprise Portal Server。

要添加 Secure Shell (SSH) 或 Windows 连接,您需要主机名、用户名和密码。

2. 添加连接时,可以从"进程浏览器"窗口的连接名称列表中选择连接。

如果不保存建立连接所需的所有字段(例如,密码),那么将打开该连接的**连接属性**窗口。输入缺失的 信息。对于 Tivoli Enterprise Portal Server Managed System 连接,必须先连接到 Tivoli Enterprise Portal Server,然后才能进入受管系统。

3. 输入用户名和密码,单击刷新(闪电)图标进行连接,然后再选择受管系统。

### 下一步做什么

要删除连接,请选择连接,然后单击编辑以打开"连接属性"窗口。选中移除此连接复选框,并单击确定。

### 监视 Windows 服务

您可以定义用于监视 Windows 系统上运行的一个或多个服务的数据源。这些服务与代理程序运行在同一主机上。对于每个服务,数据源都会向"可用性"数据集添加一行。

### 过程

1. 在"代理程序初始数据源"页面或"数据源位置"页面中,单击监视数据类别区域中的进程。

- 2. 在数据源区域中,单击 Windows 服务。
- 3. 单击下一步。
- 4. 在"服务监视器"页面的显示名字段中,输入描述。在服务名称字段中,提供服务应用程序的名称。您可以手动输入,也可以单击浏览以查看本地系统或远程系统上当前正在运行的服务列表。如果您单击浏览,则会打开"服务浏览器"窗口。此窗口初始状态下包含有关 Agent Builder 系统上每个服务的详细信息。此信息中包含服务的服务名称、显示名称、状态和描述。
**注:** 如果 Agent Builder 不是在 Windows 系统上运行,那么不会显示本地服务。必须定义或选择远程 Windows 系统,请参阅(<u>第 57 页的『定义用于服务浏览的连接』</u>)。

**注:** 当您通过 Tivoli Enterprise Portal Server 或从 UNIX 或 Linux 系统进行浏览时,服务描述不可用。 5. 选择一个或多个服务,或者执行以下一个或多个步骤来使用"**服务浏览器**"窗口中的列表:

- · 要对服务列表进行排序,请单击列标题。
- · 要刷新窗口中的信息,单击刷新(闪电)图标。
- · 要搜索服务,请单击**搜索**(望远镜)图标以打开"**服务搜索**"窗口。您可按服务名称、显示名和描述 进行搜索。
- · 要查看另一系统上的服务,请从**连接名称**列表中选择先前定义的系统,或者单击**添加**以输入系统信息。有关更多信息,请参阅<u>第57页的『定义用于服务浏览的连接』</u>。您可以同时从多个系统装入服务并且在为一个或多个连接装入服务时切换连接。

6. 在选择或输入服务名称之后,完成下列其中一个步骤:

- ·如果您正在使用"代理程序"向导,请单击下一步。
- · 单击完成,以保存该数据源并打开 Agent Editor。

#### 下一步做什么

如果想要在 IBM Cloud Application Performance Management 的摘要仪表板中使用来自此数据源的数据, 必须创建基于"可用性"数据集过滤的数据集(属性组),并将其配置为提供单行。使用"名称"字段来选 择用于您的进程的行。

在新的过滤属性组中,选择"功能测试状态"字段并为其指定严重性值。

有关指示信息,请参阅:

- ·第155页的『创建过滤属性组』
- ·第38页的『指定用作状态指示符的特性的严重性』
- · 第 187 页的『第 12 章 为 Cloud APM 准备代理程序』

## 定义用于服务浏览的连接

除了选择来自正在运行 Agent Builder 的系统的服务以外,您还可以选择来自其他 Windows 系统的服务。

## 关于此任务

要从其他 Windows 系统选择服务,请定义与远程系统的连接。您必须具有系统凭证,否则这些系统必须由 Tivoli Monitoring 操作系统代理程序监视。

### 过程

1. 要定义连接,请单击"服务浏览器"窗口中的添加。

"选择连接类型"窗口随即打开。要添加 Managed System 连接,需要 Tivoli Enterprise Server 主机 名、Tivoli Monitoring 用户名和密码以及受管系统名称。选定受管系统后,该表将列出远程系统上的服 务。

**注:** 操作系统代理程序必须在您尝试浏览的系统上运行,还必须连接到正在运行的 Tivoli Enterprise Monitoring Server 和 Tivoli Enterprise Portal Server。

添加 Windows 连接需要主机名、用户名和密码。

2. 选择连接类型(Windows 或 Tivoli Enterprise Portal Server Managed System),或者选择现有连接以用 作模板。

"连接属性"窗口随即打开。

- 3. 填写"连接属性"。
- 4. 单击完成。
- 5. 添加连接时,可以从"**服务浏览器**"窗口的连接名称列表中选择连接。

如果没有保存建立连接所需的字段(例如,密码),将打开连接属性窗口,您可以输入缺少的信息。

- a) 对于 Tivoli Enterprise Portal Server Managed System 连接,您必须先连接到 Tivoli Enterprise Portal Server,然后才能进入受管系统。输入用户名和密码,单击**刷新**(闪电)图标进行连接,然后再选择 受管系统。
- 6. 要删除连接,请执行以下步骤:
  - a) 在"服务浏览器"窗口中选择连接。
  - b) 单击编辑以打开"连接属性"窗口。
  - c) 选中移除此连接复选框。
  - d) 单击确定。

# 监视来自 Windows Management Instrumentation (WMI) 的数据

您可以定义数据源以从代理程序运行所在的系统或远程系统上的 Windows Management Instrumentation (WMI) 收集数据。数据源监视单个 WMI 类,并将此类中的所有值放在它所产生的数据集中。如果类提供多个实例,那么数据集包含多行;您可以按实例名称进行过滤以确保数据集仅包含一行。

### 开始之前

如果代理程序使用 Windows Management Instrumentation (WMI) 从远程系统收集数据,那么它需要具有访问该远程系统上的 WMI 数据的许可权。当您提供了具有远程系统上 WMI 数据的访问许可权的帐户凭证时,代理程序可以访问该系统上的 WMI 数据。管理员帐户具有必需的许可权。在随后过程中,您可以提供管理员凭证或具有必需许可权的其他用户的凭证。有关创建具有 WMI 数据浏览许可权的用户帐户的更多信息,请参阅第 178 页的『创建具有 Windows Management Instrumentation (WMI) 权限的用户』。

要通过 Windows API 收集度量值,代理程序必须在 Windows 操作系统上托管。远程系统上必须启用了远程 注册表管理。

## 过程

1. 在"代理程序初始数据源"页面或"数据源位置"页面中,单击来自服务器的数据区域中的日志数据。 2. 在数据源区域中,单击 WMI。

- 3. 单击**下一步**。
- 4. 在 "Windows Management Instrumentation (WMI) 信息"页面上,完成下列其中一个步骤:
  - · 在这些字段中输入 WMI 名称空间的名称和 WMI 类名的名称。然后,转至步骤 第 58 页的 『9』
  - · 单击浏览来查看系统上的所有 WMI 类。

要浏览远程系统,请从列表中选择系统(如果已定义)。或者,单击添加以添加 Windows 系统主机 名。提供具有远程系统上 WMI 数据的访问许可权的用户帐户凭证,或提供远程系统的管理员凭证。 此页面将更新以显示远程系统的信息。仅当在 Windows 系统上运行 Agent Builder 时才能进行浏 览,并且只能浏览 Windows 系统。

- 5. 单击类旁边的加号 (+) 以展开类并显示特性。
- 6. 从列表中,选择类以及您要为其指定的关联特性,然后单击确定。

**注:**您可以单击**搜索**(望远镜)图标,以查找列表中的所选项。请在**搜索短语**字段中输入短语;通过单 击**按名称搜索、按类描述搜索**或按类属性搜索字段指定首选项;然后单击确定。如果找到所搜索的项, 请将其选中,然后单击确定。

此时会再次打开向导的"WMI 信息"页面,其中显示所选的 WMI 类信息。

- 7. 可选: 您可以通过单击**测试**来测试此属性组。有关测试的更多信息,请参阅<u>第 59 页的『测试 WMI 属</u>性组』
- 8. 可选: 通过单击高级,您可以创建过滤器,以限制此属性组所返回的数据。有关过滤来自属性组的数据 的更多信息,请参阅<u>第 38 页的『过滤属性组』</u>
- 9. 单击下一步。

**注:** 如果手动输入了 WMI 名称空间和 WMI 类名,则您将转到"**特性信息**"页面,您可以在这里填写特性信息。在"**特性信息**"页面上,如果要添加更多特性,您可选择**添加其他特性**。单击**完成**以完成操作。

- **10**. 在"选择键特性"页面上,选择键特性或指示此数据源仅生成一个数据行。有关更多信息,请参阅<u>第</u> <u>15 页的『选择键属性』</u>。
- 11. 执行下列其中一个步骤:
  - ·如果您正在使用"代理程序"向导,请单击下一步。
  - · 单击完成,以保存该数据源并打开 Agent Editor。
- 12. 您可以添加特性并提供其相关信息。有关更多信息,请参阅 第 32 页的『创建属性』。

WMI 数据源的"特性信息"页面除了包含适用于所有数据源的字段(<u>第 35 页的表 5</u>)以外,还包含以下字段:

度量名称

要收集的类的属性名

13. 如果要为数据源设置全局选项,请单击全局选项。

如果要包括此选项,请选中包括远程 Windows 配置属性复选框,并单击确定。

有关 Windows 数据源的 Windows 远程连接配置的信息,请参阅(<u>第 177 页的『配置 Windows 远程连</u>接』)。

## 测试 WMI 属性组

如果是在 Windows 系统上运行 Agent Builder,那么可以在 Agent Builder 中测试 WMI 属性组。

### 过程

1. 可以通过以下方法启动测试过程:

- · 在创建代理程序期间,单击"WMI 信息"页面上的测试。
- · 在创建代理程序之后,在 Agent Editor 的"数据源定义"页面上选择属性组,然后单击测试。有关 Agent Editor 的更多信息,请参阅第 17 页的『第 4 章 使用 Agent Editor 修改代理程序』。

在上述两个步骤中的某个步骤中单击测试之后,将显示"WMI测试"窗口。

- 2. 可选: 在开始测试之前, 您可以设置环境变量和配置属性。有关更多信息, 请参阅<u>第191页的『属性组</u> 测试』。
- 3. 单击启动代理程序。

此时将显示一个窗口,表明代理程序正在启动。

4. 要模拟对代理程序数据的监视环境请求,请单击收集数据。

代理程序会查询 WMI 以获取数据。"WMI 测试"窗口会收集并显示自代理程序上次启动以来,代理程序高速缓存中的全部数据。

5. 可选: 如果返回的数据未如预期, 请单击检查结果。

"数据收集状态"窗口将会打开,并显示有关该数据的更多信息。<u>第 239 页的『Performance Object</u> Status 节点』中描述了"数据收集状态"窗口收集和显示的数据。

6. 通过单击停止代理程序停止代理程序。

7. 单击确定或取消以退出"WMI测试"窗口。单击确定将保存您所作的全部更改。

### 相关概念

<u>第191页的『在Agent Builder 中测试代理程序』</u>

# 监视 Windows 性能监视器 (Perfmon)

您可以定义数据源以从 Windows 性能监视器 (Perfmon) 收集数据。数据源监视 Perfmon 对象。该对象中的 计数器放置在所产生数据集中的属性中。如果类提供多个实例,那么数据集包含多行;您可以按实例名称进 行过滤以确保数据集仅包含一行。

## 过程

- 1. 在"代理程序初始数据源"页面或"数据源位置"页面中,单击来自服务器的数据区域中的日志数据。
- 2. 在数据源区域中,单击 Perfmon。
- 3. 单击下一步。
- 4. 在"Perfmon 信息"页面上,请完成下列其中一个步骤:
  - · 在对象名称字段中输入对象名称,然后单击下一步以定义属性组中的第一个属性。
    - 注:如果您为 Windows 性能监视器对象输入名称,那么此名称必须是英语名称。
  - · 单击浏览以查看 Perfmon 对象列表。

当性能监视器 (Perfmon) 的"对象浏览器"窗口最初打开时,此窗口中填充了来自本地系统的信息。 要浏览远程系统,请从列表中选择系统(如果已定义),或者单击**添加**以添加 Windows 系统主机 名。提供管理员标识和密码。此窗口将更新以包含远程系统的信息。仅当在 Windows 系统上运行 Agent Builder 时才能进行浏览,并且只能浏览 Windows 系统。例如,不能添加 Linux 或 Solaris 系统的主机名来执行远程浏览。

- 您单击对象名称时,该对象中的可用计数器将显示在窗口中。
  - 要对 Windows 性能监视器 对象或计数器进行排序,请单击列标题。
  - 要刷新窗口中的信息,请单击刷新。
  - 要搜索特定的对象或计数器,请单击**搜索**(望远镜)图标,以打开"**性能监视器搜索**"窗口。您可以搜索对象名称和/或计数器名称。搜索操作执行子串匹配,并且不区分大小写。
  - 选择对象, 然后单击确定。
  - "Perfmon 信息"页面将打开,其中的对象名称字段会显示选定对象的名称。
- ·如果要为数据源设置全局选项,请单击全局选项。

如果要包括此选项,请选中包括远程 Windows 配置属性复选框,并单击确定。

有关 Windows 数据源的 Windows 远程连接配置的信息,请参阅(<u>第 177 页的『配置 Windows 远</u>程连接』)。

- 5. 如果所选的 Windows 性能监视器对象返回多个实例,并且您希望根据实例名称来过滤结果:
  - a) 在"Perfmon 信息"页面上,选中按 Perfmon 实例名称过滤复选框。
  - b) 在 实例名称字段中,输入要过滤的实例名称,或者单击浏览以列出可用的实例。
  - c) 要浏览远程系统,从列表中选择一个系统,或者单击**添加**以添加 Windows 系统主机名。选择主机之后,请提供管理员标识和密码。该表将更新以包含远程系统上的实例列表。

注: 您还可以按属性组进行过滤, 请参阅步骤 第 61 页的『9』

6. 如果所选的 Windows 性能监视器对象要返回多个实例,并且您希望返回实例名称,请选择 "**Perfmon** 信息"页面上的返回实例名称。

选中该选项将会向数据源中添加一个属性,该属性不会显示在属性列表中。该属性包含实例名称。

注:如果浏览过选定对象,并且该对象定义为具有多个实例,那么将自动选中该复选框。

- 7. 如果未选中用于返回实例名称的选项,那么将打开"选择键属性"页面。在"选择键属性"页面上,选择键属性或指示此数据源仅生成一个数据行。有关更多信息,请参阅<u>第15页的『选择键属性』</u>。
- 8. 可选: 您可以通过单击**测试**来测试此属性组。有关测试的更多信息,请参阅<u>第 61 页的『测试</u> Perfmon 属性组』

9. 可选:通过单击高级,您可以创建过滤器,以限制此属性组所返回的数据。 有关过滤来自属性组的数据的更多信息,请参阅步骤<u>第38页的『过滤属性组』</u>

注:您还可以按实例名称进行过滤,请参阅 第 60 页的 [5]

- 10. 执行下列其中一个步骤:
  - ·如果您正在使用"新建代理程序"向导,请单击下一步。
  - · 单击完成,以保存该数据源并打开 Agent Editor。

Agent Editor 的"数据源定义"页面会显示一个列表,其中包含对象以及对象的相关信息。

11. 您可以添加属性并提供其相关信息。有关更多信息,请参阅<u>第32页的『创建属性』</u>。

数据源的"Perfmon 属性信息"页面除了包含适用于所有数据源的字段以外,还包含以下字段: 度量名称

特定对象的计数器的名称。

## 下一步做什么

有关 Perfmon 数据源的 Windows 远程连接配置的信息,请参阅<u>第 177 页的『配置 Windows 远程连接』</u>。

## 测试 Perfmon 属性组

如果您在 Windows 系统上运行 Agent Builder,可以测试所创建的 Perfmon 属性组。

## 过程

1. 可以通过以下方法启动测试过程:

- · 在创建代理程序期间,单击"Perfmon 信息"页面上的测试。
- · 在创建代理程序之后,在 Agent Editor 的"数据源定义"页面上选择属性组,然后单击测试。有关 Agent Editor 的更多信息,请参阅第 17 页的『第 4 章 使用 Agent Editor 修改代理程序』。

在上述两个步骤中的某个步骤中单击测试之后,将显示"Perfmon测试"窗口。

- 2. 可选: 在开始测试之前, 您可以设置环境变量和配置属性。有关更多信息, 请参阅<u>第191页的『属性组</u> 测试』。
- 3. 单击启动代理程序。此时将显示一个窗口,表明代理程序正在启动。
- 要模拟来自监视环境的代理程序数据请求,请单击收集数据。
   代理程序会查询性能监视器以获取数据。 "Perfmon 测试"窗口会收集并显示自代理程序上次启动以 来,代理程序高速缓存中的全部数据。

**注:**您可能直到再次单击**收集数据**之后,才会看到所有属性的有用数据。原因是某些性能监视器属性返回的是变化量值,因此需要上一个值才能计算变化量值。

5. 可选: 如果返回的数据未如预期,请单击检查结果。

"数据收集状态"窗口将会打开,并显示有关该数据的更多信息。<u>第 239 页的『Performance Object</u> Status 节点』中描述了"数据收集状态"窗口收集和显示的数据。

- 6. 通过单击**停止代理程序**停止代理程序。
- 7. 单击确定或取消以退出"Perfmon测试"窗口。单击确定将保存您所作的全部更改。

## 相关概念

<u>第191页的『在 Agent Builder 中测试代理程序』</u> 使用 Agent Builder 创建代理程序之后,可以在 Agent Builder 中测试该代理程序。

# 监视来自简单网络管理协议 (SNMP) 服务器的数据

您可以定义数据源以监视 SNMP 服务器。数据源监视来自单个 SNMP 对象标识 (OID) 和单个主机的所有数据。如果选择注册其他对象的 OID 注册树下的某个元素,将针对每组不同的标量或表值创建数据集。如果对象返回标量数据,那么数据集只包含一行。如果对象返回表格数据,那么数据集将包含多行。

## 关于此任务

代理程序支持简单网络管理协议 V1、V2C(请注意,此版本是 V2C 而不只是 V2)和 V3。

#### 过程

## 1. 在"代理程序初始数据源"页面或"数据源位置"页面中,单击来自服务器的数据区域中的日志数据。

- 2. 在数据源区域中,单击 SNMP。
- 3. 单击下一步。

4. 在"简单网络管理协议 (SNMP) 信息"页面上,输入显示名称,或者单击**浏览**以查看系统上的所有对象。

定义数据源之后,可以添加属性。这些属性的 OID 可能很长且难以正确输入。使用 Browse 选项可方便 地输入正确的 OID。

注:浏览器不浏览活动系统,而是读取定义,即管理信息库 (MIB)。

**注:**单击**刷新**图标可以清除已解析 MIB 文件的内存中版本,并重新解析工作空间高速缓存中的文件。高速缓存位于以下位置: *workspace\_directory*\.metadata\.plugins\ com.ibm.tivoli.monitoring.agentkit\mibs

其中:

#### workspace\_directory

标识在初始运行 Agent Builder 时指定的工作空间目录,请参阅(<u>第 9 页的『启动 Agent</u> Builder』)。

- a) 如果未装入定义所需对象的 MIB, 单击管理定制 MIB 以打开"管理定制 MIB"对话框。
- b) 单击添加以浏览到要添加的 MIB 文件。要从高速缓存删除 MIB,请将其选中,并单击移除。
- c) 单击确定以更新高速缓存。

如果解析 MIB 时发生任何错误,那么"管理定制 MIB"对话框保持打开状态。该对话框允许您添加 或移除 MIB 以消除错误。

单击取消会将 MIB 高速缓存恢复到打开此对话框时的状态。

Agent Builder 包含一组 MIB:

- ∙hostmib.mib
- •rfc1213.mib
- ·rfc1243.mib
- ·rfc1253.mib
- •rfc1271.mib
- ·rfc1286.mib
- ·rfc1289.mib
- •rfc1315.mib
- •rfc1316.mib
- ·rfc1381.mib
- ·rfc1382.mib
- ·rfc1443.mib
- ·rfc1461.mib
- 1101401.0010
- ·rfc1471.mib
- •rfc1493.mib
- ·rfc1512.mib
- •rfc1513.mib

•rfc1516.mib

- •rfc1525.mib
- •rfc1573a.mib
- •rfc1595.mib
- ·rfc1650.mib
- •rfc1657.mib
- •rfc1659.mib
- ·rfc1666.mib
- ·rfc1695.mib
- •rfc1747.mib
- •rfc1748.mib
- ·rfc1757.mib
- ·rfc1903.mib
- •rfc1907.mib
- ·rfc2011.mib
- ·rfc2021.mib
- ·rfc2024.mib
- ·rfc2051.mib
- •rfc2127.mib
- •rfc2128.mib
- •rfc2155.mib
- ·rfc2206.mib
- ·rfc2213.mib
- ·rfc2232.mib
- •rfc2233.mib
- •rfc2238.mib
- ·rfc2239.mib
- •rfc2320.mib
- •rfc3411.mib

所有这些 MIB 都是由 IETF 定义的标准 MIB。包括这些 MIB 的原因是,它们表示对监视可能有用的 公共定义。此外,还有许多 MIB 也是必需的,这样定制 MIB 才能解析所导入的符号。

d) 从列表中选择对象。

单击对象旁的加号(+)以展开并显示层次。

- e) 从列表中,选择要指定的对象,然后单击确定。
  - 然后,新数据源将列示在"数据源定义"页面上。

**注:** 如果选择了定义其他对象(嵌套在第一个对象下的各个对象)的对象,那么所有这些对象都将变成数据源。如果选择高级对象,那么会添加许多数据源。

- 5. 在"简单网络管理协议信息"页面上,选择操作系统。
- 6. 可选: 您可以通过单击"简单网络管理协议信息"页面上的测试来测试一个或多个数据源。 有关测试的更多信息,请参阅第 65 页的『测试 SNMP 属性组』
- 7. 可选: 通过单击高级,您可以创建过滤器,以限制此属性组所返回的数据。有关过滤来自属性组的数据 的更多信息,请参阅<u>第 38 页的『过滤属性组』</u>

8. 单击下一步。

- 9. 在"属性信息"页面上,指定属性信息。
- 10. 执行下列其中一个步骤:
  - ·如果您正在使用"新建代理程序"向导,请单击下一步。
  - · 单击完成,以保存该数据源并打开 Agent Editor。
- 11. 有关添加属性和提供属性信息的更多信息,请参阅 第 32 页的『创建属性』。

SNMP 数据源的"属性信息"页面除了包含适用于所有数据源的字段以外,还包含以下字段:

度量名称

任意字符串

#### Object identifier

注册给对象的完整 OID, 不包括索引值

## 下一步做什么

您可以使用代理程序运行时配置来设置受监视主机。

为使 Agent Builder 能够生成 64 位数据类型,并能够为 32 位无符号的 MIB 属性处理最大值,请参阅 <u>第</u>64 页的『SNMP MIB 解析选项』。

## SNMP MIB 错误

处理 SNMP MIB 中的错误。

在添加 SNMP MIB 时,经常会发现错误。单击"Agent Builder 错误"窗口中的详细信息 >> 可查看 MIB 错误详情。

一个最常见的错误是缺少在其他 MIB 中所作的定义。可以同时导入多个 MIB 以解决此问题,也可以逐次添加 MIB,直到所有缺少定义问题全部解决。Agent Builder 可以使用已解决相关问题的任何定义。因此,如果某些错误仅影响 MIB 中不打算使用的那一部分,您可以选择忽略此类错误。MIB 的顺序无关紧要,因为先将它们全部装入,然后才解析引用。

## SNMP MIB 解析选项

设置 SNMP MIB 的解析首选项

## 过程

- 1. 在 Agent Builder 中,选择窗口 > 首选项以打开"首选项"窗口。
- 2. 在导航窗格中,展开 IBM Agent Builder。
- 3. 单击 MIB 解析以打开"MIB 解析"窗口。

Agent Builder 使用的 MIB 解析器使用 ASN.1 定义的语法来解析 MIB。某些 MIB 未正确遵循该语法。解 析器允许放宽特定规则以适应最常见的错误。通过放宽这些规则,您可以解析不合规的 MIB。

### 允许类型以小写字母开头

允许用户在 MIB 中编写的类型,例如值

**允许以数字命名的编号** 允许以大写字母开头的编号

**允许在值名称中使用下划线** 允许使用下划线字符

## 允许值以大写字母开头

允许以大写字母开头的值

## 忽略重复的 MIB

关闭重复 MIB 模块警告

4. 可选:如果选中针对 32 位无符号 MIB 属性创建 64 位特性复选框,Agent Builder 就能够生成 64 位数 据类型来处理 32 位无符号 MIB 属性的最大值。选择此选项不会更改任何现有代理程序字段定义。您必 须浏览至 MIB 文件才能为这些属性创建新的数据源。 5. 完成首选项编辑时,请单击确定。

## 测试 SNMP 属性组

您可以对在 Agent Builder 中创建的 SNMP 属性组进行测试。

### 过程

- 1. 可以通过以下方法启动测试过程:
  - · 在创建代理程序期间, 单击"简单网络管理协议信息"页面上的测试。

注:

如果选择的 SNMP 对象包含多个属性组,系统将提示您选择要测试的属性组。

· 创建代理程序之后,在 Agent Editor 的"数据源定义"页面上选择属性组,然后单击测试。有关 Agent Editor 的更多信息,请参阅第 17 页的『第 4 章 使用 Agent Editor 修改代理程序』

在上述其中一步中单击测试之后,将打开"SNMP测试设置"窗口。

- 2. 从**连接名称**中选择现有连接,或者单击**添加**,然后系统将提示您选择连接类型。另外,也可使用"**创建** 连接向导"来选择现有连接以用作模板。
- 3. 在选择连接类型或现有连接之后,单击**下一步**以填写 SNMP 连接属性。完成后,单击**完成**以返回到 "SNMP 测试设置"窗口。
- 4. 可选: 在开始测试之前, 您可以设置环境变量和配置属性。有关更多信息, 请参阅<u>第191页的『属性</u><u>组测试』</u>。
- 5. 单击启动代理程序。此时将显示一个窗口,表明代理程序正在启动。
- 6. 要模拟来自 Tivoli Enterprise Portal 或 SOAP 的代理程序数据请求,请单击**收集数据**。代理程序会查询 已配置的 SNMP 连接以获取数据。
- 7. "测试设置"窗口会收集并显示自代理程序上次启动以来,代理程序高速缓存中的全部数据。
- 8. 可选:如果返回的数据未如预期,请单击检查结果。
   "数据收集状态"窗口将会打开,并显示有关该数据的更多信息。<u>第 239 页的『Performance Object Status 节点』</u>中描述了"数据收集状态"窗口收集和显示的数据。
- 9. 通过单击停止代理程序停止代理程序。
- 10. 单击确定或取消以退出"测试设置"窗口。单击确定将保存您所作的全部更改。

### 相关概念

<u>第191页的『在Agent Builder 中测试代理程序』</u> 使用 Agent Builder 创建代理程序之后,可以在 Agent Builder 中测试该代理程序。

## 监视来自简单网络管理协议事件发送方的事件

您可以定义数据源以从 SNMP 陷阱和通知事件收集数据。您必须在代理程序运行时配置中设置端口,并配置服务器以通过此端口向代理程序主机发送事件。所有受监视的事件都作为行放置在数据集中。

## 关于此任务

代理程序支持简单网络管理协议 (SNMP) V1、V2C(请注意,此版本名称是 V2C 而不只是 V2)和 V3。代理 程序可以接收和处理 SNMP 陷阱和通知。此提供者接收到的数据将作为事件传递到监视环境。

有关 SNMP 事件属性组的更多信息,请参阅(<u>第 265 页的『"SNMP 事件"属性组』</u>)。

#### 过程

1. 在"代理程序初始数据源"页面或"数据源位置"页面中,单击来自服务器的数据区域中的日志数据。

2. 在数据源区域中,单击 SNMP 事件。

- 3. 单击下一步。
- 4. 在"简单网络管理协议事件信息"窗口中,执行下列其中一个步骤:

- · 单击所有事件以创建属性组,用来为任何收到的 SNMP 事件发送事件。
- · 单击一般事件以创建属性组,用来为与任何所选通用事件类型匹配的所接收到的任何通用 SNMP 事件发送事件。
- · 单击**定制事件**以创建一个或多个属性组,用来为特定于企业的 SNMP 事件发送事件。单击**浏览**以选择要监视的事件。

在"简单网络管理协议 (SNMP) 管理信息库 (MIB) 浏览器"窗口中,按照事件定义所在的 MIB 模块 组织选择窗格中的事件。展开 SNMP 对象以显示该 MIB 模块中的事件。在列表中,单击要指定的对象,然后单击确定。

如果某个陷阱配置文件包含陷阱的静态数据,请选中**包括用于显示陷阱配置文件中所定义信息的属** 性复选框。有关 SNMP 陷阱配置文件的更多信息,请参阅(<u>第 321 页的『附录 I SNMP 陷阱配</u> 置』)。

如果要将属性以及收到的所有变量绑定 (VarBind) 数据包括在陷阱协议数据单元 (PDU) 中,请选中包括变量绑定 (VarBind) 数据属性复选框。有关此属性的更多信息,请参阅属性定义(<u>第 265 页的</u> 『"SNMP 事件"属性组』)。

注:

- a. 浏览器不浏览活动系统;而是读取定义和管理信息库 (MIB)。<u>第 61 页的『监视来自简单网络管理</u> <u>协议 (SNMP) 服务器的数据』</u>中定义了 Agent Builder 所包含的 MIB 列表。任一 SNMP 数据提供 程序装入的 MIB 在两者中都可使用。
- b. 如果选择 MIB 模块或个别事件,那么该模块中的所有事件都将转换为单独的数据源。为事件中定义的每个变量添加一个属性。如果您希望所选模块或陷阱的所有事件都到达单个事件源,请选中收集单一属性组中的事件复选框。如果选择单个陷阱,并且选择了收集单一属性组中的事件标记,那么会为每个事件中定义的每个变量(忽略重复变量)添加一个属性。如果您选择模块,那么不会添加变量属性。
- c. 如果要输入自己的过滤器, 请使用以下语法:

OID(对象标识)元素的值用于确定为此属性组处理哪些陷阱。

- **陷阱匹配**: global\_snmp\_event\_settings\_for\_group 元素的 OID 属性可以是逗号定界的标记列 表。单个标记语法如下:

[enterpriseOID][-specificType]

- 示例: "1.2.3.5.1.4,1.2.3.4.5.6.7.8.9-0"。第一个标记与企业 OID 为 1.2.3.5.1.4 的任何陷阱匹配。第二个标记与企业 OID 为 1.2.3.4.5.6.7.8.9 且特定 OID 为 0 的任何陷阱匹配。由于这些标记共同列示在同一个属性组中,因此与任一标记匹配的已接收事件都会由该属性组处理。
- d. 接收到的每个事件仅由与接收到的事件匹配的第一个属性组处理。先处理子节点属性组,然后再处理基本属性组。代理程序开发者必须确保以正确的方式定义组以便在预期属性组中接收事件。
- 5. 在"SNMP 事件信息"窗口中,选中子节点主机匹配复选框以将事件与子节点匹配。如果 SNMP 事件属 性组是子节点的一部分,可以选中子节点主机匹配复选框来控制事件是否必须来自受监视的 SNMP 代理 程序。

例如:您已让一个代理程序监视路由器,其中每个子节点实例代表一个特定路由器。您开发了一个使用 SNMP 数据收集器从路由器收集数据的代理程序。您还定义了一个接收该路由器所发送的 SNMP 事件的 属性组。每个路由器实例都包含为事件过滤器定义的相同数据。因此,您需要其他方法来确保来自路由 器的事件显示在该路由器的属性组中。

选中 Subnode Host Matching 时,由路由器发送的事件将与为 SNMP 数据收集器定义的主机进行比较。 如果 SNMP 数据收集器正在使用的主机与发送所接收事件的主机相同,那么该子节点实例会处理 SNMP 事件。否则,会将事件传递到下一个子节点实例。地址匹配仅适用于子节点。基本代理程序中的 SNMP 事件属性组不执行任何地址匹配。要使用地址匹配,子节点定义必须包含至少一个 SNMP 属性组。 SNMP 用于该子节点实例的 SNMP 主机是用于匹配的主机。

如果取消选中**子节点主机匹配**复选框,您的子节点实例不会执行这一额外的比较。必须允许用户为此情况中的各个子节点配置其他 OID 过滤器。否则,您无需在子节点定义中包含 SNMP 事件属性组。

6. 在"SNMP 事件信息"窗口中,选择操作系统。

7. 可选: 您可以单击 "SNMP 事件信息" 窗口中的测试以启动并测试代理程序。

有关更多信息,请参阅第 69 页的『测试 SNMP 事件属性组』

8. 可选:

在"SNMP 事件信息"窗口中,单击高级,以选择事件过滤和摘要选项。有关更多信息,请参阅<u>第 219</u>页的『第 16 章 事件过滤和摘要』。

a) 完成选择事件过滤和摘要选项之后,返回到"SNMP 事件信息"窗口。如果先前在"SNMP 事件信息"窗口中选择了定制事件,请单击下一步以选择键属性,否则跳过下一步。

b) 在"选择键属性"页面上,为属性组单击一个或多个键属性,或者单击生成单一数据行。

9. 单击**下一步**,如果使用"新建代理程序"向导来保存代理程序,请单击**完成**,并打开 Agent Editor。 10.

#### 下一步做什么

有关添加更多属性的信息,请参阅(第32页的『创建属性』)。

## SNMP 事件配置属性

将 SNMP 事件属性组添加到代理程序时,自动创建某些配置属性

添加数据源之后,配置将显示在 Agent Editor 的"运行时配置信息"页面上。例如,<u>第68页的图2</u>显示了配置节,以及在将 SNMP 事件属性组添加到代理程序时自动创建的一些配置属性。

😓 *Agent Editor Project One 🛛 🧼 Remote Deploy Bundle Editor									
Runtime Configuration Information	<u></u>								
Runtime Configuration Information									
Custom Configuration     Bernard Configuration     Configuration for Simple Network Management Protocol (SNMP)     Description     SNMP Events	Add								
123 Port Number									
<ul> <li>Security Level</li> <li>User Name</li> <li>Auth Protocol</li> <li>Auth Password</li> <li>Priv Password</li> <li>Trap configuration file</li> <li>Configuration for Simple Network Management Protocol (SNMP)</li> <li>Configuration for Java Virtual Machine (JVM)</li> <li>Configuration for Java Virtual Machine (JVM)</li> <li>Configuration for Java Database Connectivity (JDBC)</li> <li>Subnode configuration</li> </ul>									
Runtime Configuration Details Information about the configuration property	^								
Label Port Number									
Environment variable KQZ_SNMPEVENT_PORT	th label								
Description The port number used to listen for SNMP events									
Type Numeric	~								
Default value 162 Multiple	Values								
Choices									
E Ren	lit lit								
Agent Information   Data Sources Runtime Configuration   itm_toolkit_agent.xml									

## 图 2. "运行时配置"页面

预定义配置属性的标签、描述和缺省值可以更改,但变量名称和类型不能更改。SNMP 事件配置节包含以下属性:

表 10. SNMP 事件配置属性					
姓名	有效值	是否必需	描述		
端口号	正整数	是	用于侦听事件的必需端口 号		
安全级别	noAuthNoPriv、 authNoPriv 和 authPriv	否	SNMP V3 安全级别		
用户名	字符串	否	SNMP V3 用户名		
认证协议	MD5 或 SHA	否	SNMP V3 认证协议		
认证密码	字符串	否	SNMP V3 认证密码		

表 10. SNMP 事件配置属性 (续)					
姓名	有效值	是否必需	描述		
专用密码	字符串	否	SNMP V3 隐私密码		
陷阱配置文件	包括路径的文件名	否	陷阱配置文件的位置。如 果使用此配置属性未找到 该文件,会尝试在代理程 序 bin 目录查找 trapcnfg 文件。		

V1 或 V2C 事件不需要配置。无论是指定源名称还是共用名,都将处理所有 V1 或 V2C 事件。唯一支持的隐私协议为 DES,所以没有指定隐私协议的选项。没有必需的 SNMP V3 配置选项(可以选择指定任何选项)。如果要指定这些选项,那么必须为所选的安全级别指定适当的值。

## 测试 SNMP 事件属性组

您可以对在 Agent Builder 中创建的 SNMP 事件属性组进行测试。

## 开始之前

要测试 SNMP 事件属性组,请使用测试程序,或用于生成 SNMP 事件的应用程序。

## 过程

1. 可以通过以下方法启动测试过程:

- · 在创建代理程序期间, 单击 "SNMP 事件信息" 窗口中的测试。
- · 在创建代理程序之后,在 Agent Editor 的"数据源定义"页面上选择属性组,然后单击测试。有关 Agent Editor 的更多信息,请参阅<u>第 17 页的『第 4 章 使用 Agent Editor 修改代理程序』</u>

在上述两个步骤中的某个步骤中单击测试之后,将打开"测试事件设置"窗口

- 2. 可选: 在开始测试之前,您可以设置环境变量和配置属性。有关更多信息,请参阅<u>第191页的『属性组</u> 测试』。有关 SNMP 事件配置属性的更多信息,请参阅 <u>第 67 页的『SNMP 事件配置属性』</u>。
- 3. 单击**启动代理程序**。此时将显示一个窗口,表明代理程序正在启动。 当代理程序启动时,它会根据其配置侦听 SNMP 事件。

注: 启动的代理程序是简化的版本, 其中包含要测试的一个属性组。

4. 要测试代理程序的数据收集,需生成与代理程序配置匹配的 SNMP 事件。可以使用应用程序或事件生成器执行此操作。

当代理程序接收到与其配置匹配的 SNMP 事件时, 会将这些事件添加到其内部高速缓存。

5. 要模拟来自监视环境的代理程序数据请求,请单击**收集数据**。

"测试事件设置"窗口会收集并显示自代理程序上次启动以来,代理程序高速缓存中的全部事件。<u>第</u>70页的图3中显示了示例数据收集

en started. Log files								
ion searced, bog nes	can be found in	C:\Users\mtruss	\AppData\Local	\Temp\KQZ_1328875551075	i\TMAITM6\l	ogs.		
~		Start A	gent Co	llect Data Stop Agent	Ched	k Results S	et Environment	Configuration
urce Address	Generic Trap	Specific Trap	Alert Name	Event Variables	Category	Description	Enterprise Name	Severity
cm-9-67-222-100	1	3	_	{1.3.18[Counter32]=34}				
cm-9-67-222-100	1	3		{1.3.18[Counter32]=34}				
cm-9-67-222-100	1	3		{1.3.18[Counter32]=34}				
cm-9-67-222-100	1	3		{1.3.18[Counter32]=34}				
cm-9-67-222-100	1	3		{1.3.18[Counter32]=34}				
.cm-9-67-222-100	1	3		{1.3.18[Counter32]=34}				
e U 9 9 9	25 rrce_Address cm-9-67-222-100 cm-9-67-222-100 cm-9-67-222-100 cm-9-67-222-100 cm-9-67-222-100	Ince_Address         Generic_Trap           cm-9-67-222-100         1           cm-9-67-222-100         1           cm-9-67-222-100         1           cm-9-67-222-100         1           cm-9-67-222-100         1           cm-9-67-222-100         1	Start A           Ss           rce_Address         Generic_Trap         Specific_Trap           cm-9-67-222-100         1         3           cm-9-67-222-100         1         3	Start Agent         Co           35         Generic_Trap         Specific_Trap         Alert_Name           cm-9-67-222-100         1         3         cm-9-67-222-100         1         3           cm-9-67-222-100         1         3         cm-9-67-222-100	Start Agent         Collect Data         Stop Agent           25         generic_Trap         Specific_Trap         Alert_Name         Event_Variables           cm-9-67-222-100         1         3         {1.3.18[Counter32]=34}           cm-9-67-222-100         1         3         {1.3.18[Counter32]=34}	Start Agent         Collect Data         Stop Agent         Check           25	Start Agent         Collect Data         Stop Agent         Check Results         S           25         3         41.3.18[Counter32]=34]         Category         Description           cm-9-67-222-100         1         3         {1.3.18[Counter32]=34}         Cm-9-67-222-100         I <td>Start Agent         Collect Data         Stop Agent         Check Results         Set Environment         C           25         3         1.3.18[Counter32]=34]         Enterprise_Name         Enterprise_Name           cm-9-67-222-100         1         3         {1.3.18[Counter32]=34}         Enterprise_Name           cm-9-67-222-100         1</td>	Start Agent         Collect Data         Stop Agent         Check Results         Set Environment         C           25         3         1.3.18[Counter32]=34]         Enterprise_Name         Enterprise_Name           cm-9-67-222-100         1         3         {1.3.18[Counter32]=34}         Enterprise_Name           cm-9-67-222-100         1

### 图 3. 显示所收集 SNMP 事件数据的"测试事件设置"窗口

## 6. 可选: 如果返回的数据未如预期,请单击检查结果。

"数据收集状态"窗口将会打开,并显示有关该数据的更多信息。(<u>第 70 页的图 4</u>)中显示了示例。 第 239 页的『Performance Object Status 节点』中描述了"数据收集状态"窗口收集和显示的数据。

🔁 Data Collec	tion Status		·						×
Data Collect	ion Status								
Collection stat	us for testing attribu	ute group Traps							
Query_Name	Object_Name	Object_Type	Object_Status	Error_Code	Last_Collection_Start	Last_Collection_Finished	Last_Collection_Duration	Average_Collection_[	Duration
Traps	SNMP Events: *	SNMP_EVENT	ACTIVE	NO_ERROR	01-Jan-1970 00:00:00	01-Jan-1970 00:00:00	0	NO DATA	
-									
<u> </u>									
0									OK I
									OK

## 图 4. "数据收集状态" 窗口

7. 通过单击停止代理程序停止代理程序。

8. 单击确定或取消以退出"测试事件设置"窗口。单击确定将保存您所作的全部更改。

## 相关概念

<u>第191页的『在Agent Builder 中测试代理程序』</u> 使用 Agent Builder 创建代理程序之后,可以在 Agent Builder 中测试该代理程序。

# 监视 Java 管理扩展 (JMX) MBean

您可以定义数据源以从 JMX MBean 收集数据。将来自于每个受监视的 MBean 的数据放置在一个数据集中。 根据 MBean,数据集可生成一行或者多行。

## 关于此任务

您定义的每个 JMX 数据源都必须标识单个 MBean(单个实例)或特定类型的 MBean(多个实例)。您必须 知道 MBean 的对象名称或一种 MBean 类型(包含要收集的数据)的对象名称模式。对象名称模式只能用于 标识一组相似的 MBean。与该模式匹配的这组 MBean 必须都能提供您要在监视表中查看的数据。典型的对 象名称模式类似于 \*:j2eeType=Servlet,\*。此对象名称模式与具有 j2eeType 的 servlet 的所有 MBean 相匹配。您可能希望与该模式匹配的任何 MBean 都具有一组类似的、可添加到数据源的公开特性和操作。 使用该模式的数据源将从与该模式匹配的每个 MBean 收集数据。您为此数据源定义的特性必须可供与此数 据源对象名称模式相匹配的任何 MBean 使用。 支持 Java V5 或更高版本。

## 过程

1. 在"代理程序初始数据源"页面或"数据源位置"页面中,单击来自服务器的数据区域中的日志数据。

- 2. 在数据源区域中,单击 JMX。
- 3. 单击下一步。
- 4. 在"JMX 信息"页面上,单击浏览以查看 MBean Server 上的所有 JMX MBean。

定义数据源之后,可以使用 Browse 功能预填充特性列表。然后可以添加、移除或修改浏览器插入的特性。这些特性的名称可能很长且难以正确输入。使用 Browse 选项可方便地输入正确的名称。

**注:**您可以手动创建 JMX 数据源,方法是指定对象名称并单击下一步,而无需使用浏览器。手动创建 JMX 数据源将创建两个数据源。将创建一个包含 JMX 通知的预定义特性的事件数据源。另外,还将定 义集合数据源,其中包含必须在向导中指定的一个特性。

#### MBean pattern

显示 MBean 模式。

**Global JMX Options** 

显示支持级别。

为以下 JMX 服务器提供了支持:

- ·Java 5 操作系统 MBean Server。通过使用 JSR-160 接口建立连接。支持通知和监视器。
- ·WebSphere<sup>®</sup> Application Server V6 和更高版本。对 SOAP 和 RMI 协议都提供了接口。由于远程 代理程序无法创建 MBean,因此不支持 JMX 监视器。
- ·WebSphere Community Edition 和其他基于 Apache Geronimo 的应用程序服务器。通过标准 JSR-160 接口建立连接。JMX 通知和监视器在 V1.1 和更高版本中受支持。
- ·JBoss Application Server V4.0 和更低版本。
- ·JBoss Application Server JSR-160 连接。
- ·WebLogic Server V9 和更新版本。为 T3 协议提供了接口。
- 5. 在首次运行 JMX Browser 时, **MBean Server** 向下滚动菜单中没有任何项。要添加连接,请单击**添加**按 钮。

使用**编辑**按钮可修改或删除您已定义并在向下滚动菜单中选中的连接。连接定义存储在工作空间中,这 样在创建连接时就会记住该定义。完成以下步骤来创建连接。如果您已有连接,请跳至下一步。

a) 要创建与 MBean Server 的连接,请单击添加以添加连接,或者编辑现有连接。

如果尚未定义任何连接,此时将显示"Java 管理扩展 (JMX) 浏览器"窗口。

- b) 在单击添加以添加连接之后,将打开"选择连接类型"页面。
- c) 使用"MBean Server 连接"向导来连接到 MBean Server。页面上所列的新连接是为创建连接可以选择的选项。您可以将现有连接作为模板,使用现有连接列表来创建新连接。选择其中一种新连接类型,然后单击**下一步**以开始创建连接。
- d) 在选择连接类型之后,可能会要求您选择更具体的连接类型。此时将显示基于标准 JMX 连接 (JSR-160)连接类型的两个模板。选择最合适您的 MBean Server 的模板,然后单击下一步。

🔁 Create Connec	tion Wizard						
Connection Pro	operties						
Edit the connectio	n properties and press Finish.						
Connection name	JBoss JSR-160						
JMX user ID							
JMX password							
	Save the password in the Agent Builder workspace	e					
JMX service URL	service:jmx:remoting-jmx://localhost:9999						
- lava class nath in	formation						
IMX base naths	C:\iboss-eap-6.3.01\iboss-eap-6.3	Browse					
JHX base patris							
JMX class path	Din\client\jboss-client.jar	Browse					
JMX JAR directorie	JMX JAR directories Browse						
Browser Java Rur	time Environment						
Java location C:	\Program Files (x86)\IBM\Java70\jre	Browse					
	Test Connection						
Set as agent configuration defaults							
		1					
(?)	< Back Next > Finish	Cancel					

图 5. JMX 连接属性

"**连接属性**"页面(<u>第72页的图5</u>)包含有关如何连接到 MBean Server 的详细信息。您必须将 MBean Server 的相关详细信息填写到该页面中。

**要点:** 如果数据源连接到远程 WebSphere Application Server, 请确保 WebSphere Application Server 也安装在运行 Agent Builder 的主机上,并将 **Java 位置**设置设定为本地 WebSphere Application Server 使用的 Java 运行时环境。

- e) 如果要保存此连接的密码,请选中将密码保存在 Agent Builder 工作空间中复选框。
- f) 可选:如果要从这些连接属性中复制 JMX 缺省值,请选中设置为代理程序配置缺省值。
   例如,在<u>第72页的图5</u>中,缺省的 JMX 基本路径是 C:\jboss-eap-6.3.01\jboss-eap-6.3, JMX 服务 URL 是 service:jmx:remoting-jmx://localhost:9999,而 Java 位置是 C:\Program Files\IBM\Java70\jre

- 1) 在指定连接所需的属性之后,请单击**测试连接**以确保可以建立此连接。如果连接失败,请更正必 需的属性。
- 2) 连接成功后,请单击完成以返回到使用所配置连接的浏览器。

"连接属性"页面中的 Java 类路径信息包含三个字段。必须根据需要填写这三个字段才能连接到其所 需 Java 类不在 Java 运行时环境中的 MBean Server。通常,您要连接的 MBean Server 必须与 Agent Builder 安装在同一系统上。在此情况下,请在 JMX 基本路径字段中指定包含 MBean Server 的应用程 序的安装目录。然后,JMX Jar 目录字段将列出"基本路径"目录的相对目录,这些目录包含连接到 MBean Server 所需的 JAR 文件。JMX 类路径字段可以用于包含特定 JAR 文件。JMX JAR 目录字段中 所列的 JAR 文件不需要单独列示在 JMX 类路径字段中。

任意字段都可包含多个引用;以分号分隔条目即可。这些值是配置代理程序时所需的值。有关更多信息,请参阅第 77 页的『JMX 配置』。

6. 选择连接后, JMX Browser 将从 JMX 服务器下载有关 MBean 的信息。此信息显示在"JMX 浏览器" 窗口的以下四个区域中(第 74 页的图 6):

从"Java 管理扩展 (JMX) 浏览器"窗口开始到 Agent Editor 的运行时配置选项卡的屏幕方向:从 "JMX 信息"页面中,选择浏览。从浏览器 (未选择连接的"JMX 浏览器")中,选择添加。从 "JMX 连接选择"页面中,选择 JBoss,然后选择下一步。从"JMX 连接属性"页面中,定制以下两 个连接属性: JBoss 提供程序 URL 的值为 jnp://wapwin3.tivlab.raleigh.ibm.com:1099/, JBoss Jar 目录 的值为以下 JAR 文件所在目录的完整路径: jbossall-client.jar、jbossjmx.jar、jboss-jsr77-client.jar 和 jboss-management.jar。选择完成。此配置将设置 JBoss 连接,这样您会看到类似以下所示的屏幕。

Bean serve	r JBoss JSR-1	.60					▼ Add	Edit	N
MBean Key	Properties				name Va	lues			
[Dom     subsy     extens     type     horne     name	ain] stem ion tq-server		)	•	Module Servicel class sto default direct jboss-as ibors in	LoaderInt ModuleLo orage s	egration-7 ader-5		• III •
2eeTupe	subsystem	extension	tune	bornet	n-cen/er	name	Other Key Propert	ier	
2EEType DEEServer	subsystem	extension	type	nome	iq-server	default	other key Propert		
IVM						default	J2EEServer=defau	lt	
Class N Descrip MBea Nam	lame: org.jb btion: Mana in Attributes e	oss.as.jsr77.n gement Obje MBean Ope Description	nanaged ect rations	lobject. MBea	J2EEServer n Notificat Type	Handler	Read/Write		
objec	:tName 'Ms	The object of The java vm	name 15	iype java.lang.String [Ljava.lang.Stri		Read Only Read Only		- []	

图 6. Java Management Extensions (JMX) Browser 窗口

- · MBean 键属性区域:此区域是从服务器上所有 MBean 中发现的所有唯一对象名称键的集合。 [Domain] 是特殊条目,因为它并不是键。但是,[Domain] 条目被视为 MBean 域值的隐式键。从此 列表选择项,并找到包含该键属性的 MBean。键属性值列表显示在选择的键属性值列表中。当您勾选 一个键属性后,它将包含在数据源的对象名称模式中。
- ·选择的键属性值区域:此区域显示当前从所有 MBean 中选择的"MBean 键属性"值。选择其中一个值也将勾选对应的 MBean 键属性。此选择还会将此 MBean 键属性的名称和值更新到消息字段中显示的对象名称模式。

· 一个列出所有与对象名称模式匹配的 MBean 的表: 当您从 MBean Key Properties 和 Selected Key Property Values 列表中选择键属性和值后,您将看到对象名称模式更新。您还将看到该表中的 MBean 列表更改为反映与选定模式匹配的 MBean 列表。如果具有与任何 MBean 都不匹配的模式,那 么您可以清除 MBean Key Properties 列表中的条目。通过单击模式所使用的键旁边的复选框并移除复 选标记来清除条目。您也可以手动编辑模式,以查找需要的 MBean。模式 \*:\* 将选择所有 MBean。

您可以使用此表来浏览服务器中的 MBean, 然后确定哪些包含了您要监视的数据。要帮助浏览可能非 常多的 MBean,可以按任意键特性排序(使用菜单或单击列标题)。您还可以从菜单中选择显示键属 性来显示任意列中的任意键特性。当您在表中看到标识了要监视的 MBean 的键属性时,右键单击该 值并从菜单中选择**仅选择具有键属性的 MBean**。

·包含有关选定 MBean 的详细信息的表: JMX Browser 向您显示单个 MBean 的相关信息。要查看某个 MBean 的详细信息,请从显示与当前过滤器匹配的 MBean 列表的表中选择该 MBean。有关 MBean 的键信息是其定义的特性、操作和通知的列表。

要从 JMX Browser 创建数据源,可使用之前描述的四个面板来构建对象名称模式。构建与一组 MBean (其中每个 MBean 都包含您要收集的监视数据)匹配的对象名称模式。例如,如果要监视来自所有 ThreadPool MBean 的数据,请使用以下步骤:

- a) 从"**MBean 键属性**"面板选择**类型**。选择**类型**将导致更新"**选择的键属性值**"中的值,列出任意 MBean 的此类型键的所有唯一值。
- b) 从此类型键的值列表中选择**线程池**。选择**线程池**之后,将在"**MBean 键属性**"面板中选中此类型键 属性名称,并且对象名称模式将更新为\*:type=ThreadPool,\*。MBean 列表也将更新,以仅显示与该 模式匹配的 MBean。
- c) 从 MBean 列表选择一个 MBean, 以查看对该 MBean 可用的特性、操作和通知。如果 MBean 列表 包含的 MBean 超出要监视的数量,那么您必须继续执行选择键属性和值的这一过程。继续该过程, 直到您得到可识别要监视的 MBean 集合的对象名称模式。您也可以在 MBean 列表中打开一个菜 单,以使用表中显示的键属性值更新对象模式。
- 7. 当对象名称模式正确时,从表选择一个 MBean。

选定 MBean 的所有特性在新的 JMX 数据源中作为初始特性。某些特性可能不包含数据。创建 JMX 数据 源后,请复审特性并移除所有不太重要的特性。如果选定的 MBean 没有任何特性,将警告您即将创建 没有任何特性的数据源。如果选定的 MBean 包含通知,那么还将创建事件数据源以接收来自 MBean 的 通知。

**要点:** 对于每个 MBean 特性, Agent Builder 会在新数据集中创建一个特性。对于数字 MBean 特性, Agent Builder 会创建一个数字特性。对于任何对象类型,包括 String, Agent Builder 将创建一个字 符串特性,其中包含值的字符串表示。如果 MBean 特性中对象的类型为

javax.management.openmbean.CompositeData,而且 Agent Builder 浏览器可读取该对象本身,那么它会创建多个特性,即 CompositeData 对象中嵌入的每个对象都有一个相应的特性。要包括非 CompositeData 对象的对象内部的值(字段或方法返回值),您需要创建一个具有更复杂度量名称的特性,如<u>第 83 页的『Java 管理扩展(JMX) MBean 的特定字段』</u>所述。

8. 在填写完毕的"JMX 信息"页面中,单击完成。

数据源是根据在上一步中选定的 MBean 创建的。如果未选择任何 MBean,那么将创建一个没有任何特性的属性组。此时将显示警告,以允许您选择 MBean。通知数据源的数据源名称开头含有 Event 一词,用以与显示特性的数据源进行区分。

- 9. 要更改代理程序的其他 JMX 选项,可单击全局 JMX 选项。使用这些选项,您可以执行以下操作:
  - a) 选择此代理程序是否支持 JMX 监视器。如果要创建 JMX 监视器属性组和"执行操作"命令,请选择 包括 JMX 监视器属性组及执行操作。

请参阅下一章节以获取 JMX 监视器的描述。

- b) 选择代理程序部署时要连接的 MBean 服务器类型。 将列出若干特定于供应商的服务器类型,以及基于标准的服务器的通用 JSR-160 兼容服务器。您可以根据需要选择多个类型,但必须只选择支持要监视的 MBean 的服务器类型。必须至少选择一个类型。如果选择多个类型,那么将在配置代理程序时提示您指定要连接的服务器类型。
- 10. 选择所需选项之后,单击确定。
- 11. 可选: 您可以通过单击**测试**来测试此属性组。有关测试的更多信息,请参阅(<u>第 86 页的『测试 JMX</u> 属性组』)
- 12. 可选:通过单击**高级**,您可以创建过滤器,以限制此属性组所返回的数据。有关过滤来自属性组的数据 的更多信息,请参阅<u>第 38 页的『过滤属性组』</u>
- 13. 单击下一步。
- 14. 在"选择键属性"页面上,选择键属性或指示此数据源仅生成一个数据行。有关更多信息,请参阅<u>第</u> 15页的『选择键属性』。
- 15. 单击下一步。

"JMX 代理程序范围的选项"窗口将显示 Agent Builder 支持的应用程序服务器的类型。如果之前在 "连接属性"页面上选择了设置为代理程序配置缺省值,那么将自动选择您所浏览到的应用程序服务器 的类型。

16. 在"JMX 代理程序范围的选项"窗口(<u>第 76 页的图 7</u>)中,选择您希望代理程序能够连接到的任何其他类型的应用程序服务器。

注: 在所示示例中,选择 JBoss Application Server JSR-160 连接等同于选择符合 JSR-160 标准的服务器,不同之处在于所提供的缺省值不同。

23 IBM Tivoli Monitoring Agent Component Wizard
JMX Agene-Wide Options
Select options for the JMX attribute group.
✓ Include JMX monitor attribute groups and take actions.
Select the server configuration choices you would like to be available when the agent is deployed.
<ul> <li>Standard JMX Connections (JSR-160)</li> <li>JSR-160-Compliant Server</li> <li>WebSphere</li> <li>WebSphere Application Server version 6.0</li> <li>WebSphere Application Server version 7.0 and newer</li> <li>WebSphere Application Server Community Edition (JSR-160)</li> <li>IBoss</li> <li>JBoss Application Server version 4 and earlier</li> <li>JBoss Application Server JSR-160 connection</li> <li>WebLogic</li> <li>WebLogic Server version 9</li> <li>WebLogic Server version 10 and newer</li> </ul>
(?) < <u>Back</u> <u>Next</u> > <u>Finish</u> Cancel

图 7. JMX Agent-Wide Options 窗口

- 17. 执行下列其中一个步骤:
  - · 如果您正在使用"新建代理程序"向导,请单击下一步。
  - · 单击完成,以保存该数据源并打开 Agent Editor。
- 18. 在创建代理程序之后,如果要更改可以连接的应用程序服务器类型,请单击 JMX 数据源信息区域中的 全局 JMX 选项。
- 19. 在"JMX 代理程序范围的选项"窗口中,更改所需的任何选择。
- 20. 单击确定。
- 21. 要查看自动生成的配置节和属性,可单击 Agent Editor 的运行时配置选项卡。

JBoss 基本路径属性的缺省值与 JMX Browser 中输入的值相同。

## 下一步做什么

有关 JMX 事件属性组的更多信息,请参阅 第 266 页的『"JMX 事件"属性组』。

## JMX 配置

在代理程序中定义 JMX 数据源时,将为您创建部分配置属性。

JMX 运行时配置是唯一的,因为它使您能够对所显示配置量进行一些控制。代理程序的 JMX 客户机可以连接到多种不同类型的应用程序服务器。但是,无需在任一代理程序中支持所有类型的应用程序服务器。您可以确定要支持哪些类型的应用程序服务器,且在代理程序中将不包含不必要的配置节。

在大多数情况下,代理程序设计为监视一种 JMX 应用程序服务器类型。在创建 JMX 数据源,可以使用 JMX Browser。在使用 JMX Browser 时,会自动向代理程序添加用于浏览 MBean Server 的 JMX 服务器配置选项。在创建代理程序之后,要更改可以连接的应用程序服务器类型,请单击"JMX 信息"区域中的全局 JMX 选项。在"JMX 代理程序范围的选项"页面中,更改所需的任何选择。

您可以设计用于监视多种类型的 JMX 应用程序服务器的通用代理程序。在此情况下,可以在"JMX 代理程 序范围的选项"页面上选择多个 JMX 服务器配置选项。如果支持多种 JMX 连接类型,那么运行时配置会提 示您选择将用于代理程序实例的连接类型。

**注:** 一个代理程序实例只能连接一种 JMX 应用程序服务器。在一个代理程序实例内,子节点可用于连接相同 类型的不同 JMX 应用程序服务器。要连接多种 JMX 应用程序服务器,必须为每种 JMX 应用程序服务器类型 配置至少一个代理程序实例。

您可以使用 Agent Editor 来查看、添加和更改配置属性。有关指示信息,请参阅<u>第 177 页的『使用 Agent</u> Editor 更改配置属性』。如果在子节点中定义了 JMX 数据源,那么还能指定"子节点配置覆盖"。有关指示 信息,请参阅<u>第 165 页的『子节点配置』</u>。

如果在代理程序中定义了 JMX 数据源,那么该代理程序必须使用 Java 来连接 JMX 应用程序服务器。Java 配置属性将自动添加到该代理程序中。

下列 Java 配置属性特定于代理程序运行时配置:

#### Java 主目录

指向 Java 安装目录的标准路径

将代理程序配置为使用的 JVM 与您所监视的应用程序使用的 JVM 相同,对于 WebLogic Server 和 WebSphere Application Server 尤其应这样配置。

#### JVM 自变量

向 Java 虚拟机指定可选自变量列表。

#### 跟踪级别

定义要写入 Java 跟踪文件的信息量。缺省值为只将错误数据写入日志文件。

注: Agent Builder 不需要这些属性,因为它使用自己的 JVM 以及日志记录,这是通过 JLog 插件配置的。

如果在代理程序中定义 JMX 数据源,那么将向代理程序中自动添加以下必需的公共配置字段:

连接

与 MBean Server 的连接的类型

### 用户标识

用于向 MBean Server 认证的用户标识。

#### 密码

用户标识的密码。

#### 基本路径

用于在其中搜索**类路径**所指定 JAR 文件的目录,或者是 JAR 目录中指定的目录(非标准目录)。在 Windows 上使用分号(;)分隔目录名称,在 UNIX 系统上使用分号(;)或冒号(:)分隔目录名称。

#### 类路径

代理程序将搜索的显式指定的 JAR 文件。在找到 JAR 文件之前,所有非标准文件都将追加到每个 Base Paths 中。

### JAR 目录

要在其中搜索 JAR 文件的目录。在 Windows 上使用分号(;)分隔目录名称,在 UNIX 系统上使用分号(;)或冒号(:)分隔目录名称。不需要显式标识这些目录中的 JAR 文件;之所以找到这些文件是因为它 们确实在其中一个目录中。但不会搜索这些目录的子目录。在找到目录之前,所有非标准目录名都将追 加到每个 Base Paths。

**注:** 对于远程监视,这些 JAR 文件及其所有从属 JAR 文件必须以本地方式安装在代理程序运行所在的计算机 上。这些 JAR 文件是连接到受监视应用程序所必需的文件。这些 JAR 文件必须在 JAR 目录中配置,并且必 须位于**基本路径**和**类路径**中。此外,以本地方式安装所监视应用程序支持的 JVM,并在 Java 主目录配置字 段中指定路径。

### 示例:

· 对于 WebLogic 10, 类路径为 server/lib/wlclient.jar; server/lib/wljmxclient.jar。基本 路径指向 server/lib 目录所在的 WebLogic 应用程序服务器目录。

·对于 WebSphere,基本路径指向 WebSphere Application Server 的安装位置。此示例中列出了多个基本路径,为 Windows 和 UNIX 提供了缺省值。类路径列出了相对于基本路径的 JAR 文件。JAR 目录字段的相对值 lib 导致装入基本路径下此目录中的所有 JAR 文件。

- 基本路径: C:\Program Files\IBM\WebSphere\AppServer;/opt/IBM/WebSphere/ AppServer
- 类路径: runtimes/com.ibm.ws.admin.client\_6.1.0.jar;plugins/ com.ibm.ws.security.crypto\_6.1.0.jar
- JAR 目录: lib

根据在 JMX Agent-Wide Options 页面中选择的 JMX 服务器类型,将添加以下部分或全部配置属性。缺省值 将由 Agent Builder 提供,且可以修改:

### 特定于 JSR-160 兼容服务器连接的配置属性:

#### JMX 服务 URL

要连接以进行监视的 JMX 服务 URL。

### 特定于 WebSphere Application Server V6.0 及更高版本连接的配置属性:

#### 主机名

要监视的应用程序服务器所在系统的主机名。对于本地监视,此名称是本地系统名称。对于远程监视,此名称是应用程序服务器所在的系统的主机名。

## 端口

在要监视的主机名中使用的端口号。

#### 连接器协议

监视连接要使用的接口协议。支持 RMI 和 SOAP。

#### 概要文件名称

用于配置连接的概要文件的名称。

#### 特定于 JBoss Application Server(非 JSR-160)连接的配置属性:

#### JNDI 名称

用于查找 MBean Server 的 JNDI 名称。

#### 提供程序 URL

要连接以进行监视的 JMX 服务提供者 URL。

### 特定于 WebLogic Server 连接的配置属性:

#### 服务 URL

要连接以进行监视的 JMX 服务提供者 URL(包含 JNDI 名称)。

**注:** 如果启用了 WebSphere 管理安全,那么必须确保在相应的客户机连接属性文件中禁用客户机登录提示。 对于 RMI 连接,要阻止客户机提示用户,必须修改 WebSphere Application Server 概要文件属性目录中 sas.client.props 文件内的 com.ibm.CORBA.loginSource 属性。对于 SOAP 连接,必须修改同一目录中 soap.client.props 文件内的 com.ibm.SOAP.loginSource 属性。在这两种情况下, loginSource 属性都必须设置为不包含值。

您可以使用 Agent Editor 来查看、添加和更改配置属性。请参阅(<u>第 177 页的『使用 Agent Editor 更改配置</u> 属性』)。如果子节点中定义了 Windows 数据源,您还可以指定"子节点配置覆盖"。请参阅<u>第 165 页的</u> <u>『子节点配置』</u>。

## JMX 通知

除了应请求提供监视数据外,一些 MBean 还提供通知。

通知是由 MBean 所生成的对象,在发生事件时传递到注册侦听器。

Agent Builder 所构建的代理程序可以定义一些属性组以包含来自通知(而非 MBean)的值。

当代理程序启动后,通知侦听器会注册到与属性组的 MBean 模式匹配的所有 MBean。然后,属性组将显示 收到的通知,每行显示一个通知。每列将包含一个来自通知的数据项。来自通知的所需数据由列值定义,方 法与为 MBean 定义列数据类似。

对于并非基于事件的属性组,将在需要时收集数据。对于基于事件的属性组,代理程序将维护最近收到的 100个事件的高速缓存。这些事件用于响应来自 Tivoli Enterprise Portal 的请求。这些事件将立即转发,以 供情境进行分析以及进行仓储。

## JMX 监视器

除了应请求提供监视数据外,一些 MBean 还提供监视。

JMX 提供者支持代理程序创建 JMX 监视器的能力。JMX 监视器是 JMX 代理程序在 JMX 服务器上创建的 MBean。它监视其他 MBean 的属性值并在该值符合某些条件时发送通知。定义的阈值使监视器能报告特定 属性值。

并非所有应用程序服务器都支持从 JMX 客户机创建监视器,这对于当前发行版的 WebSphere Application Server 也是如此。可以将 JMX 监视器和"执行操作"命令包括在代理程序中,方法是选择"全局 JMX 选项"下的包括 JMX 监视器属性组及执行操作。

任何报告其他 MBean 属性的 MBean 都可视为监视器。实际上, JMX 定义三个具体的监视器类, 这三个类都 是创建的监视器类型。将创建以下具体的监视器类型:

·字符串监视器 - 观察字符串属性, 报告字符串是否相等。

- ·标尺监视器 观察可变数字属性, 报告超出阈值的上溢或下溢活动。
- ·计数器监视器 观察增大数字属性, 当达到阈值或增加了特定量时进行报告。

以下属性组可自动添加到代理程序,以收集或表示 JMX 监视器通知:

·已注册的监视器

此属性组显示用户添加的所有 JMX 监视器。

·计数器通知

此属性组报告从计数器监视器收到的所有通知。

·标尺通知

此属性组报告从标尺监视器收到的所有通知。

·字符串通知

此属性组报告从字符串监视器收到的所有通知。

### 针对 JMX 监视器的"执行操作"命令

监视器通过运行"执行操作"命令创建。

将定义三个"执行操作"命令,每一个命令用于创建一种类型的监视器,还将定义第四个"执行操作",用于删除现有监视器。对"执行操作"命令应用了 256 个字符的限制。

监视器属性组是所构建的每个 JMX 代理程序(包括 Agent Builder 构建的所有代理程序)的一部分。这四个"执行操作"命令对所有代理程序都可用,但是它们不能用于非 JMX 代理程序。

#### JMX 添加字符串度量观察程序

使用此"执行操作"命令来创建监视器以观察字符串属性。

#### 参数

#### MBean 模式

与此模式匹配的所有 MBean 受此监视器监视。

#### 受观测属性

所观察的 MBean 字符串属性的名称。

#### 通知匹配

当受监视字符串与参考值匹配时,如果将发送通知,那么为 true,否则为 false(缺省值为 false)。

#### 通知差异

当受监视字符串与参考值不匹配时,如果将发送通知,那么为 true,否则为 false(缺省值为 true)

#### 参考值

要与受观测属性进行比较的字符串。

缺省值表示未指定自变量。

#### 示例:请求当服务停止时进行通知

STRING\_METRIC\_WATCHER [\*:type=Service,\*] [StateString] [true] [false] [Stopped]

其中:

#### \*:type=Service,\*

MBean 模式: 监视具有键属性指定类型值 Service 的任意 MBean。

#### StateString

受观测属性:对所有类型为 Service 的 MBean 通用的字符串属性。

#### true

通知匹配:您希望在 StateString 属性与参考值 Stopped 匹配时,向代理程序发送通知。

#### false 译面句

通知差异:您不希望在 Service 属性与 Stopped 不匹配时收到通知。

### Stopped

参考值:当 StateString 属性更改为值 Stopped 时,发送通知。

### JMX 添加标尺度量观察程序

使用此"执行操作"命令来创建监视器以观察标尺属性。

## 参数

### MBean 模式

与此模式匹配的所有 MBean 受此监视器监视。

## 受观测属性

所观察的 MBean 字符串属性的名称。

### 差异方式

如果受监视值等于属性的实际当前值与先前值的差,那么为 true。如果受监视值等于属性的实际当前 值,那么为 false(缺省值为 false)。

### 阈值上限通知

当增大的受监视值超出阈值上限时,如果将发送通知,那么为 true,否则为 false(缺省值为 true)。

### 阈值下限通知

当减小的受监视值超出阈值下限时,如果将发送通知,那么为 true,否则为 false(缺省值为 true)。

### 阈值上限

受观测属性应保持低于此值。

#### 阈值下限

受观测属性应保持高于此值。

#### 示例:请求可用内存低于 10 MB 时进行通知

GAUGE\_METRIC\_WATCHER [ServerInfo] [FreeMemory] [false] [false] [true] [30000000] [10000000]

### 其中:

#### \*:type=ServerInfo

MBean 模式: 监视名称中包含单个键属性指定类型值 ServerInfo 的任意 MBean。

## FreeMemory

受观测属性:上下波动的数字属性,该属性指示应用程序服务器中的可用内存量。

#### false

差异方式:监视实际属性值,而不是某一观测值与另一观测值之间的差异。

#### false

阈值上限通知:可用内存增加时不发送通知。

true 阈值下限通知:可用内存变得过低时不发送通知。

#### 3000000

阈值上限:即使您并不在意是否超过阈值上限,仍需要合理的阈值上限。在属性值达到或超过阈值上限 之前,不会再次发出阈值下限通知。

### 1000000

阈值下限:希望得到有关通知的阈值下限。

#### JMX 添加计数器度量观察程序

使用此"执行操作"命令来创建监视器以观察计数器属性。

### 参数

#### MBean 模式

与此模式匹配的所有 MBean 受此监视器监视。

#### 受观测属性

所观察的 MBean 字符串属性的名称。

#### 初始阈值

用来与受观测属性比较的值。

#### 偏离量

这是超过阈值后对阈值增加的值,用来创建一个更改的阈值。

#### 模数

这是计数器的最大值,达到此值之后计数器回滚为0。

### 差异方式

如果受监视值等于属性的实际当前值与先前值的差,那么为 true。如果受监视值等于属性的实际当前 值,那么为 false(缺省值为 false)。此方式将有效开启更改率监视。

#### 频率

执行测量的频率(缺省值为 20 秒)。如果差异方式为 true, 此项将最重要

#### 示例:请求当任意服务器包含三个或以上错误时进行通知

COUNTER\_METRIC\_WATCHER [\*:j2eeType=Servlet,\*] [errorCount] [3] [4] [] [diff] [gran]

### 其中:

#### \*:j2eeType=Servlet,\*

MBean 模式:监视名称中包含单个键属性指定类型值 ServerInfo 的任意 J2EE servlet MBean

### errorCount

受观测属性:不断增大的数字属性,该属性指示 servlet 的错误数。

#### 3

初始阈值:您希望在 errorCount 达到或超过 3 时收到通知。

4

偏移量:得到出现三个错误的通知时,将对先前阈值3加4以得到新的阈值7。在 errorCount达到7时,将发送第二个通知;达到11时,将发送第三个通知;达到15时,将发送第四个通知,以此类推。0或"无"为无效值,因为预期计数器始终是增加的,而不增加偏移量对计数器没有任何意义。

#### 模数:

errorCount 没有已构造的最大值,因此使用非常大的值。

false

差异方式:您关注的是绝对错误计数。如果所关注的是 errorCount 的增加率,差异将为 true。

频率:未设置,因此使用缺省频率 20 秒。频率对所有监视器类型可用。但是,它随计数器监视器一起显示,以便能够确定有意义的更改率(差异方式 = true)。

#### JMX 删除度量观察程序

使用此"执行操作"可删除监视器。

#### 参数

#### 编号

REGISTERED\_MONITORS 表中所显示的监视器编号

#### 示例:删除2号监视器

DELETE\_WATCHER [2]

#### 其中:

2=

要删除的监视器的编号。

### JMX 操作

除了应请求提供监视数据外,一些 MBean 还提供操作。

具有 JMX 数据源的代理程序包含 JMX\_INVOKE "执行操作"命令,您可使用该命令来针对所监视的服务器运行 JMX 操作。

### "执行操作"命令语法

该操作语法如下:

JMX\_INVOKE [MBean pattern] [Operation name] [Argument 1] [Argument 2]
[Argument 3] [Argument 4]

### 其中:

#### **MBean pattern**

用来选择要对其运行操作的 MBean 的 MBean 查询。如果该模式与多个 MBean 相匹配,那么该操作在 每个匹配的 MBean 上运行。

#### **Operation name**

要运行的 MBean 操作的名称。

#### Argument 1、Argument 2、Argument 3 和 Argument 4

可以提供给 MBean 操作的可选自变量。自变量必须是简单数据类型,如字符串或整数。

如果操作成功运行,那么 JMX invoke "执行操作"命令返回"成功"。如果操作返回一个值,该值将写入 JMX 数据提供程序日志文件。

## 示例: 启动用于重置计数器的操作

此操作对 Threading MBean 运行 resetPeakThreadCount 操作:

```
JMX_INVOKE [*:type=Threading,*] [resetPeakThreadCount][] [] []
```

其中:

#### \*:type=Threading,\*

MBean 模式:此模式与类型为 Threading 的所有 MBean 相匹配。

#### resetPeakThreadCount

操作名称:在与该模式相匹配的每个 MBean 上运行的操作。

0000

自变量 1、2、3 和 4: 这些自变量并非此操作需要的自变量。指定它们只是为了符合该操作的语法。

#### 示例:带自变量启动操作

此操作对 Threading MBean 运行 getThreadCpuTime 操作。结果记录到 JMX 数据提供程序跟踪文件中。

JMX\_INVOKE [\*:type=Threading,\*] [getThreadCpuTime] [1] [] []

其中:

#### \*:type=Threading,\*

MBean 模式:此模式与类型为 Threading 的所有 MBean 相匹配

## getThreadCpuTime

操作名称:在与该模式相匹配的每个 MBean 上运行的操作。

1

自变量1: 被查询的线程标识。

#### 000

自变量 2、3 和 4: 这些自变量并非此操作需要的自变量。将它们指定为空自变量是为了符合"执行操 作"命令语法。

### 运行 JMX\_INVOKE "执行操作" 命令

代理程序开发者无法预见到用户会运行 JMX\_INVOKE "执行操作"命令。而是必须开发更多运行 JMX\_INVOKE "执行操作"的操作。如果可能,在这些操作中向用户隐藏一些详细信息,如操作名称和 MBean 模式。

### 启动和停止 JMX 监视器

JMX 监视器在启动和停止代理程序与 JMX 服务器时始终存在。

如果代理程序检测到 JMX 服务器已重新启动,将重新注册监视器。如果代理程序重新启动,也将重新注册监视器。监视器定义存储在名为 default\_instanceName.monitors 的文件中,其中 instanceName 是代理程序实例名称,如果只有一个代理程序实例,这是缺省名称。该文件位于以下目录中(请注意, xx 表示两个字符的产品代码):

- ·Windows 系统: TMAITM6/kxx/config
- ・UNIX 和 Linux 系统: *architecture/xx*/config(请参阅<u>第 206 页的『系统上的新文件』</u>以了解有关 确定体系结构值的信息)

如果重新启动代理程序,它将使用监视器定义文件来恢复监视器。

## Java 管理扩展 (JMX) MBean 的特定字段

在"特性信息"窗口上进行指定时, JMX 属性组的度量名称语法必须遵循某些规则。

JMX 属性组的度量名称语法包含以句点分隔的标记。这些标记组成了主要的值以及可选的辅助值:

· **主值**: 这是在表的特定行中从 MBean 或"通知"直接获取的值。来自 MBean 的主要值可从 MBean 特性 或通过 MBean 操作调用(方法调用)获取。来自"通知"的主要值可从"通知"对象上的字段或方法调 用获取。主要值可以是基本类型,也可以是 Java 对象。

·**辅助值**:这是通过进一步处理主要值或其他辅助值获取的值。辅助值是在引擎内部处理的,且不会涉及对 JMX 服务器的调用。如果主要值(或其他辅助值)是 Java 对象,那么辅助值将是访存该对象公共字段的 结果。辅助值还可以是该对象的方法调用的结果。此类辅助值通过使用主要(或其他辅助)Java 对象的 Java 自省获得。如果主要(或其他辅助)值是形式为 MBean 名称的 Java 字符串,那么辅助值可以是域。 辅助值还可以是组成该 MBean 名称的任意属性。

以下语法描述度量名称字段的格式:

PrimaryValue [ .SecondaryValue ] Metric Name = PrimaryValue = Attribute.attributeName | Method.methodName | Domain | Property.propertyName | Field.fieldName | Name SecondaryValue = Field.fieldName | Method.methodName Domain | Property.propertyName | Explode | ElementCount

```
propertyName = the name of a key property in an MBean ObjectName
attributeName = the name of an MBean attribute
methodName = a zero-argument operation of an MBean or a zero-argument method
of a Notification or other Java object.
methodName(argument) = A single-argument operation of an MBean or a
single-argument method of a Notification or other Java object. The
argument will be passed to the method as a string.
fieldName = the name of a public instance variable in a Notification or
other Java object
notificationMethod = the name of a public zero-argument method of a
Notification object
```

如果在度量名称定义中仅包含主要值,那么收集的数据可以是以下任意项:

- MBean 域
- ·MBean 字符串值
- ·MBean 名称的键属性
- ·MBean 特性的数字或字符串特性值(包含另一 MBean 的全名)。来自 MBean 操作的数字或字符串返回 值。
- · "通知"对象中的数字或字符串公共实例变量的值
- ·来自"通知"操作的数字或字符串返回值。

通过向度量定义添加辅助值,您可以向下钻取到 Java 对象的主要值。此外,还可以启动公共方法或访存公 共实例变量。

通过向度量定义中另一辅助值添加辅助值,您可以向下钻取到辅助值对象。MBean 或"通知"内部嵌套了多 少对象,您就可以向下钻取到所对应的深度。

组成主要值和辅助值的标记可以是关键字或名称。在大多数情况下,关键字标记后跟名称标记。下表显示了 某些示例:

度量名称样本	属性组类型	返回数据的描述
域	MBean	MBean 的域部分(冒号之前的部分)。
Name	MBean	MBean 的完整字符串表示法。
Attribute.serverVendor	MBean	MBean 特性 serverVendor。
Method.getHeapSize	MBean	MBean 上由 getHeapSize() 返回的值。
Property.j2eeType	MBean	从 MBean 名称抽取的 j2eeType 的值。

度量名称样本	属性组类型	返回数据的描述
Field.Message	事件(通知)	通知中的 <b>消息</b> 字段。

关键字 Attribute、Method 和 Field 可以返回包含其他数据的 Java 对象。您可以通过追加辅助值定义 对那些对象运行操作。更多示例:

度量名称样本	属性组类型	返回数据的描述
Attribute.deployedObject.Method.getN ame	MBean	从 MBean 获取 deployedObject 特性,并得 到 getName() 方法的结果。
Attribute.eventProvider.Method. getException.Method.getDescription	MBean	此过程分三步:假设一个名为 eventProvider 的特性是具有 getException() 方法的对象。 此方法通过 getDescription() 方法返回一个 对象。调用此方法,并将返回值放至列中。
Attribute.HeapMemoryUsage.Method. get(used)	MBean	从 MBean 获取 HeapMemoryUsage 特性, 并获取 get(String value) 方法的结果。所使 用的字符串作为自变量传递到此方法。只能 提供一个自变量,并且该自变量必须是文字 串值。 显示如何从开放式 MBean 组合数据结构收集

如果上一个值返回格式为 MBean 名称的字符串,那么 Domain 和 Property 可用作辅助值中的关键字。例 如:

度量名称样本	属性组类型	返回数据的描述
Attribute.jdbcDriver.Property .name	MBean	特性 jdbcDriver 返回 MBean 名称,键属性 name 从该 MBean 名称中抽取。
Attribute.jdbcDriver.Domain	MBean	特性 jdbcDriver 返回 MBean 名称,domain 从该 MBean 名称中抽取。

ElementCount 和 Explode 关键字对数组或数据集合运行操作。

·ElementCount-返回数组中的元素数量。

·Explode - 将一行拆分为多行,数组的每个元素对应一个新行。

每个关键字的示例:

度量名称样本	属性组类型	返回数据的描述
Attribute.deployedObjects.ElementCou nt	MBean	MBean 特性 deployedObjects 是一个数组, 且该列包含数组中元素的数量。
Attribute.deployedObjects.Explode. MBean.Property.j2eeType	MBean	将使所部署对象的每个元素在表中都有一个 对应的行。此列包含部署对象的j2eeType。
Attribute.SystemProperties.Method. values.Explode.Method.get(key)	MBean	使开放式 MBean 表格数据结构中的每个条目 都能获得一个新行。每个表格数据结构都包 含组合数据结构,此结构具有返回的名为 key 的项。

## 测试 JMX 属性组

您可以对在 Agent Builder 中创建的 JMX 属性组进行测试。

## 过程

- 1. 可以通过以下方法启动测试过程:
  - · 在创建代理程序期间, 单击 "JMX 信息"页面上的测试。
  - · 在创建代理程序之后,在 Agent Editor 的"数据源定义"页面上选择属性组,然后单击测试。有关 Agent Editor 的更多信息,请参阅第 17 页的『第 4 章 使用 Agent Editor 修改代理程序』。

在上述两个步骤中的某个步骤中单击测试之后,将显示"JMX测试"窗口

- 2. 从**连接名称**下提供的列表中选择连接,或者单击**添加**以添加连接,并执行<u>第70页的『监视 Java 管理扩展 (JMX) MBean』</u>中详细描述的过程。
- 可选: 在开始测试之前,您可以设置环境变量、配置属性和 Java 信息。
   有关更多信息,请参阅<u>第 191 页的『属性组测试』</u>。有关 JMX 配置的更多信息,请参阅 <u>第 77 页的</u> <u>『JMX 配置』</u>。
- 4. 单击启动代理程序。
  - 此时将显示一个窗口,表明代理程序正在启动。
- 5. 单击**收集数据**可模拟来自 Tivoli Enterprise Portal 或 SOAP 的代理程序数据请求。 代理程序会监视 JMX 服务器以获取数据。"JMX 测试"窗口会收集并显示自代理程序上次启动以来,代 理程序高速缓存中的全部数据。
- 6. 可选: 如果返回的数据未如预期,请单击检查结果。

"**数据收集状态**"窗口将会打开,并显示有关该数据的更多信息。<u>第 239 页的『Performance Object</u> Status 节点』中描述了 Data Collection Status 窗口收集和显示的数据

- 7. 通过单击停止代理程序停止代理程序。
- 8. 单击确定或取消以退出"JMX测试"窗口。单击确定将保存您所作的全部更改。

### 相关概念

<u>第191页的『在Agent Builder 中测试代理程序』</u> 使用 Agent Builder 创建代理程序之后,可以在 Agent Builder 中测试该代理程序。

# 监视来自公共信息模型 (CIM) 的数据

您可以定义数据源以从公共信息模型 (CIM) 数据源接收数据。数据源监视单个 CIM 类,并将此类中的所有值 放在它所产生的数据集中。如果类提供多个实例,那么数据集包含多行;您可以按实例名称进行过滤以确保 数据集仅包含一行。

## 关于此任务

此任务描述了配置公共信息模型 (CIM) 数据源的步骤。

### 过程

- 1. 在"代理程序初始数据源"页面或"数据源位置"页面中,单击监视数据类别区域中的来自服务器的数据。
- 2. 在数据源区域中,单击 CIM。
- 3. 单击下一步。
- 4. 在"公共信息模型 (CIM) 信息"页面的 CIM 信息区域中,选择下列其中一个选项:
  - · 填写名称空间和 CIM 类名字段,以指定您想要收集的数据。
  - · 单击浏览,以浏览特定系统上的 CIM 存储库。

这将显示"**公共信息模型 (CIM) 类浏览器**"窗口。此浏览器会连接至 CIM 服务器,并为您提供有关 该服务器上存在的类的信息。

要浏览远程系统,请从**主机名**列表中选择系统(如果已定义)。或者,单击**添加**以添加 CIM 服务器 所在的系统的主机名。

用于指定主机名的语法为 http[s]://hostname:port。如果您只提供了主机名,那么公共信息 模型 (CIM) 类浏览器将使用缺省 URL http://hostname:5988 进行连接。

如果您提供了协议,但未指定端口,那么将使用 5988 作为缺省值(对于 http),或使用 5989 作为缺省值(对于 https)。

如果提供了端口但未指定协议,那么 http 将与提供的端口一起使用。

提供帐户的用户标识和密码,该帐户需对您想浏览的名称空间中的对象具有读许可权。将使用远程 系统的信息对此窗口进行更新。

Agent Builder 将尝试发现 CIM 服务器上可用的名称空间。发现的名称空间将显示在**名称空间**列表中。但是, Agent Builder 可能只能发现该服务器上的部分可用名称空间。如果您想要浏览未在**名称 空间**列表中列出的名称空间,请单击**名称空间**列表旁边的加号 (+) 图标。在字段中输入该名称空间的 名称, 然后单击确定。如果 CIM 服务器上存在该名称空间,那么将列示该名称空间中定义的类。您 输入的名称空间将保存下来,并在您下一次浏览该特定 CIM 服务器时放入**名称空间**列表。

您从**名称空间**列表中选择名称空间时, Agent Builder 会收集该特定名称空间的所有类信息。然后, Agent Builder 将缓存该信息, 以使您能够在名称空间之间快速切换。如果您要强制 Agent Builder 重新收集特定名称空间的类信息, 请选择该名称空间, 然后单击**连接**。单击**连接**将删除全部已缓存的 信息, 并导致 Agent Builder 重新收集类信息。

您可以单击**搜索**(望远镜)图标,以查找列表中的所选项。请在**搜索短语**字段中输入短语;通过单击**按名称搜索**或按类属性搜索字段指定首选项;然后单击确定。如果找到所搜索的项,请将其选中,然后单击确定。

- 5. 在"公共信息模型 (CIM) 信息"页面上的操作系统区域中,选择要执行此收集的操作系统。
- 6. 如果在 CIM 信息区域中输入了名称空间和 CIM 类名,请执行下列步骤:
  - a) 单击下一步以显示"属性信息"页面,并定义属性组中的第一个属性。

b) 指定"属性信息"页面的相关信息, 然后单击完成。

- 7. 如果您已浏览 CIM 信息,那么将显示"选择键属性"页面。在"选择键属性"页面上,选择键属性或指示该数据源只生成一个数据行。有关更多信息,请参阅<u>第15页的『选择键属性』</u>。
- 8. 如果您已浏览到 CIM 信息,请单击完成。
- 9. 可选: 您可以通过单击**测试**来测试此属性组。有关测试的更多信息,请参阅<u>第88页的『测试 CIM 属性组』</u>。
- **10.** 可选:通过单击**高级**,您可以创建过滤器,以限制此属性组所返回的数据。有关对来自属性组的数据进行过滤的更多信息,请参阅<u>第38页的『过滤属性组』</u>
- 11. 执行下列其中一个步骤:
  - a) 如果您正在使用"代理程序"向导,请单击下一步。
  - b) 单击完成, 以保存该数据源并打开 Agent Editor。

### CIM 配置

有关 CIM 配置属性的详细信息。

如果在代理程序中定义 CIM 数据源, CIM 配置属性将自动添加到代理程序。您可以使用 Agent Editor 来查看、添加和更改配置属性。有关指示信息,请参阅<u>第 177 页的『使用 Agent Editor 更改配置属性』</u>。如果在 子节点中定义了 CIM 数据源,那么请指定"子节点配置覆盖"。有关指示信息,请参阅<u>第 165 页的『子节</u> 点配置』。

以下特定于连接的配置属性在 CIM 配置页面上:

#### CIM 本地或远程

向 CIM 服务器进行本地或远程认证。"本地/远程"缺省值为远程

#### CIM 用户标识

用于访问 CIM 服务器的用户标识

#### CIM 密码

用于访问 CIM 服务器的密码

### CIM 主机名

要访问以获取 CIM 数据的主机名

#### 使用 SSL 进行 CIM 通信

使用 SSL 与 CIM 服务器通信。选项为"是"和"否"。缺省值为"否"。

### CIM 端口号

用于非安全通信的端口号。

#### CIM SSL 端口号

用于安全通信的端口号。缺省值为 5989。(Solaris 8 的缺省值通常不同。)

## 测试 CIM 属性组

您可以对在 Agent Builder 中创建的 CIM 属性组进行测试。

### 过程

1. 可以通过以下方法启动测试过程:

- · 在创建代理程序期间,单击"CIM 信息"页面上的测试。
- · 在创建代理程序之后,在 Agent Editor 的"数据源定义"页面上选择属性组,然后单击测试。有关 Agent Editor 的更多信息,请参阅第 17 页的『第 4 章 使用 Agent Editor 修改代理程序』

在上述两个步骤中的某个步骤中单击测试之后,将显示"测试设置"窗口

- 可选: 请先设置环境变量及配置属性, 然后再开始测试。
   有关更多信息, 请参阅第 191 页的『属性组测试』。
- 3. 选择或添加主机名。

有关添加主机名的更多信息,请参阅第86页的『监视来自公共信息模型 (CIM)的数据』。

4. 单击启动代理程序。

这将打开一个窗口,指示代理程序正在启动。

- 5. 要模拟来自 Tivoli Enterprise Portal 或 SOAP 的代理程序数据请求,请单击**收集数据**。 代理程序将查询 CIM Server 以获取数据。"**测试设置**"窗口会收集并显示自代理程序上次启动以来,代 理程序高速缓存中的全部数据。
- 6. 可选: 如果返回的数据未如预期,请单击检查结果。

"数据收集状态"窗口将会打开,并显示有关该数据的更多信息。<u>第 239 页的『Performance Object</u> Status 节点』中描述了"数据收集状态"窗口收集和显示的数据

- 7. 通过单击停止代理程序停止代理程序。
- 8. 单击确定或取消以退出"测试设置"窗口。单击确定将保存您所作的全部更改。

## 相关概念

<u>第191页的『在Agent Builder 中测试代理程序』</u> 使用 Agent Builder 创建代理程序之后,可以在 Agent Builder 中测试该代理程序。

## 监视日志文件

您可以定义数据源以从文本日志文件接收数据。代理程序定期解析添加日志文件的行,并基于这些行生成事件信息。您可以配置代理程序将日志解析为事件的方式。您还可以配置代理程序以过滤和汇总数据。产生的 事件放置在数据集中。

### 开始之前

注:代理程序监视的日志文件与代理程序运行所使用的语言环境和代码页相同。

过程

1. 在"代理程序初始数据源"页面或"数据源位置"页面中,单击监视数据类别区域中的记录的数据。

2. 在数据源区域中,单击日志文件。

3. 单击下一步。

4. 在"日志文件信息"页面上的日志文件信息区域中,输入要监视的日志文件的名称。

- 文件名必须是标准文件名。
- a) 可选: 日志文件名的一部分可能取自运行时配置属性。要创建日志文件名, 可单击**插入配置属性**并 选择配置属性。
- b) 可选:此文件也可以是动态文件名。有关更多信息,请参阅<u>第 317 页的『附录 H 动态文件名支</u> 持』。
- 5. 在字段标识区域中,单击下列其中一个选项:

### **Fixed number of characters**

如果选择此选项,将限制字符的数量。

通过使用此选项,将为每个特性分配它可容纳的日志文件中的最大字符数。例如,如果有三个特性 A、B和C(按此顺序),并且每个特性是最大长度为20的字符串。那么日志记录的前20个字节 将进入A,接下来的20个字节进入B,再接下来的20个字节进入C。

### 制表符分隔符

如果选择此选项,那么可使用制表符分隔符。

### Space separator

如果选择此选项,那么多个连续空格将用作一个分隔符。

**Separator Text** 

如果选择此选项,需输入分隔符文本。

Begin and End Text

如果选择此选项,需输入开始和结束文本。

### XML in element

如果选择此选项, 输入要用作记录的 XML 元素的名称, 或者单击浏览以定义元素。

如果您已单击**浏览**,则会显示"**XML 浏览器**"窗口。如果使用 Browse 功能, Agent Builder 将通过 查看子标记及其特性来识别记录的所有可能特性。

注:除非单击高级并填写该窗口中的信息,否则对于您所填写的信息,将作出以下假设:

- ·一次仅监视一个日志文件。
- ·日志文件的每行都包含填写要定义的特性所必需的所有字段。

有关日志文件解析和分隔符的更多信息,请参阅(第95页的『日志文件解析和分隔符』)。

6. 可选: 在"日志文件信息"页面上单击高级,使用"高级数据源属性"页面完成以下操作:

·监视多个文件,或监视不同操作系统上不同名称的文件,或监视名称与正则表达式匹配的文件。

·从日志文件中的多行提取一组字段。

·选择事件过滤和摘要选项。

·生成输出摘要信息。该摘要在每个时间间隔都会再生成一个属性组。有关此属性组的更多信息,请参阅<u>第 250 页的『日志文件摘要』</u>。此功能已被 Event Information 选项卡中提供的选项取代。

a) 要监视多个日志文件,请单击**添加**并输入名称。

如果列出了多个文件,那么必须为每个文件输入唯一的标签。标签可作为特性显示以指示哪个文件生成了记录。标签不得包含空格。

b) 可选: 要选择待监视的每个日志文件所在的操作系统, 请执行下列步骤:

1) 单击日志文件的操作系统列。

2) 单击**编辑**。

- 3) 在"操作系统"窗口中,选择操作系统。
- 4) 单击确定以保存更改,并返回到"高级数据源属性"页面。

c) 可选:如果您提供的文件名是一个正则表达式,而该正则表达式用于查找文件,而不是用作文件名,请选择**文件名与正则表达式匹配**。

有关更多信息,请参阅<u>第 309 页的『附录 F ICU 正则表达式』</u>。如果未选中此复选框,那么此名称 必须是实际的文件名。或者,它必须是遵循<u>第 317 页的『动态文件名语法』</u>中所述文件名模式规则 的模式。

d)可选:选择一个目录元素与正则表达式匹配以将文件名路径的一个子目录与正则表达式匹配。 如果在上一步中选择了文件名与正则表达式匹配,则可同时选择此选项。

如果在路径名中使用了正则表达式元字符,那么只能在路径的一个子目录中使用这些元字符。例如,可以指定 /var/log/[0-9\.]\*/mylog.\*以在一个子目录中包含元字符。[0-9\.]\*将导致与仅包含数字和点(.)的 /var/log 任意子目录匹配。mylog.\*将导致与那些 /var/log 子目录中所有以 mylog 开头并且后跟零个或零个以上字符的任意文件名匹配。

某些操作系统使用反斜杠(\)作为目录分隔符,这可能会与正则表达式转义元字符混淆。为了避免 此类混淆,必须始终使用正斜杠来指示目录。例如,指定为C:\temp\mylog.\*的Windows文件 可能表示\t是缩写的制表符。因此,在所有操作系统上,请始终使用正斜杠(/)作为目录分隔符。 C:/temp/mylog.\*示例表示C:/temp目录中所有以mylog开头的文件。

- e) 在当多个文件匹配时列表中, 选择下列其中一个选项:
  - ·文件名中的数字值最大的文件
  - ·最大的文件
  - ·更新时间最近的文件

·创建时间最近的文件

·所有匹配的文件

注: 在选择所有匹配的文件时,代理程序将识别目录中与动态文件名模式匹配的所有文件。代理程序 以并行方式监视所有文件的更新。所有文件中的数据会在数据收集过程中混和。最好是在记录字段 信息中选择日志文件名以添加特性,从而将日志消息和包含该日志消息的日志文件关联起来。确保 与动态文件名模式匹配的所有文件都可以采用一致的方式分割成多个特性。如果无法一致地解析所 选日志文件,那么最佳做法是在记录字段信息中选择整个记录来定义单个特性。有关为特性指定记 录字段信息的更多信息,请参阅步骤 <u>第 91 页的『8』</u>。

f) 选择处理文件的方式。

通过**对文件进行采样时处理所有记录**,每次为日志监视器定义的采样时间间隔到期时,您都可以处理整个文件中的所有记录。缺省时间间隔为 60 秒。您可以使用 *KUMP\_DP\_COPY\_MODE\_SAMPLE\_INTERVAL* 环境变量(指定以秒为单位的值)来修改此时间间隔。除非从文件中移除这些记录,否则每次都会报告相同的记录。如果使用该选项,那么向文件中写入新记录时不会生成事件数据。通过**处理追加到文件中的新记录**,您可以处理在代理程序运行期间追

新记录时不会生成事件数据。通过**处理追加到文件中的新记录**,您可以处理在代理程序运行期间追 加到文件的新记录。对于添加到文件中的每个记录,都会生成一个事件记录。如果替换了文件(第 一个记录以任何方式发生更改),那么将处理文件,并且对于文件中的每条记录生成一个事件。

**注:** 如果将记录追加到 XML 日志文件,那么追加记录必须包含您选作**字段标识**的 XML 元素中定义的 一组完整元素。

g) 如果选择处理追加到文件中的新记录, 那么还可以选择检测新记录的方式。

通过记录计数增大时检测新记录,当文件中的记录数增加时,无论文件大小是否更改,都可以检测 到新记录。在向日志文件中写入任何记录之前预先分配整个文件时,该功能很有用。对于非预先分 配的文件也可以选择该选项,但与监视文件的大小相比,此选项效率较低。通过文件大小增大时检 测新记录,您可以确定何时通过典型方法向文件追加新条目。在识别受监视文件被替换时,可能会 有一个短暂的延时。

h)如果选择了文件大小增大时检测新记录,则还可以选择如何处理监视代理程序启动时存在的文件。 忽略现有记录禁止在代理程序启动时为文件中的任意记录产生事件。处理文件中的\_\_\_ 个现有记录 指定在代理程序启动时,为文件结尾处固定数量的记录产生事件。处理代理程序先前未处理的记录:针对要由监视代理程序维护的重新启动数据指定,以便代理程序知道其上次运行时处理了哪些 记录。对于自上次代理程序运行以来追加到文件中的任何记录都生成事件。每次将记录添加到文件 时,此选项都会涉及一些额外的处理。 i) 如果选择了**处理代理程序先前未处理的记录**,则可以选择在代理程序启动时明显替换了现有文件的 情况下要执行的操作。

**在文件替换后处理所有记录**:如果有关受监视文件的信息和重新启动数据信息不匹配,那么会为文件中的所有记录产生事件。不匹配的示例包括:文件名不同、文件创建时间不同、已减小文件大小、文件上次修改时间早于原先的时间。**在文件替换后不处理记录**:如果有关受监视文件的信息和重新启动数据信息不匹配,那么会禁用对文件中现有记录的处理。

j) 单击记录标识选项卡可将日志文件中的多行解释为单个逻辑记录。

**注:** 如果在"日志文件信息"页面上选择元素中的 XML 作为字段标识,那么不会显示记录标识选项 卡。

- · 单一行: 将每个行解释为单个逻辑记录。
- ·分隔行: 当您输入一串字符时, 此行标识用于分隔记录的行。
- 注:分隔符行不是上一记录或下一记录的组成部分。
- ·规则:标识构成记录的最大行数,并可以选择性地标识一串字符以指出记录的开头或结尾。通过规则,您可以指定以下属性:
- 最大非空白行数: 定义规则可以处理的最大非空白行数。
- 规则类型可以是下列其中一项:
  - **不比较文本**(每个记录的最大行数指示单个逻辑记录)。
  - 识别记录开头(标记单个逻辑记录的开头)。
  - 识别记录结尾(标记单个逻辑记录的结尾)。
- 偏移量:指定行中必须出现"比较字符串"的位置。
- 比较测试:可以是等于(要求在特定偏移位置存在字符序列匹配项)或不等于(指示在特定偏移位置不出现特定字符序列)。
- 比较字符串: 定义要比较的字符序列。
- ·正则表达式:标识用于指示记录开头或结尾的模式。通过使用正则表达式,可以指定以下属性:
- 比较字符串: 定义要匹配的字符序列。

或者

- 记录的开头或结尾:
  - 识别记录开头标记单个逻辑记录的开头。
  - 识别记录结尾标记单个逻辑记录的结尾。
- k) 如果您之前已选择**对文件进行采样时处理所有记录**,请单击**过滤表达式**选项卡。通过单击**过滤表达** 式,您可以根据一个或多个特性的值和/或配置变量的值来过滤返回为行的数据。

如果您之前已选择**处理追加到文件中的新记录**,那么无法创建过滤表达式。有关过滤来自属性组的数据的更多信息,请参阅(<u>第 38 页的『过滤属性组』</u>)。

l) 如果您之前已选择**处理追加到文件中的新记录**,请单击事件信息选项卡,以选择事件过滤和摘要选项。

有关更多信息,请参阅<u>第 219 页的『第 16 章 事件过滤和摘要』</u>。

**注:** 如果代理程序是使用先前版本的 Agent Builder 创建的,那么可能存在 Summary 选项卡。现已由 Event Information 选项卡取代了该 Summary 选项卡

- 7. 可选: 在"日志文件信息"页面上单击单击测试日志设置,以启动并测试数据源。. 在选择日志源选项 之后,请单击测试日志文件设置。在测试日志文件数据源和提供日志文件内容时,Agent Builder 会基于 解析日志的结果自动在组中创建特性。有关测试的更多信息,请参阅<u>第 96 页的『测试日志文件属性</u> 组』。
- 8. 如果之前并未使用测试功能,并且在"日志文件信息"页面上的日志文件信息区域中输入了日志文件 名,请使用以下步骤:
  - a) 单击下一步以显示"特性信息"页面,并定义属性组中的第一个特性。
  - b) 在"特性信息"页面上指定信息, 然后单击完成。

**注:** 在缺省的 Tivoli Monitoring 最低版本 (6.2.1) 或更高版本中向代理程序添加了日志文件属性组时,将 包含日志文件状态属性组。有关日志文件状态属性组的更多信息,请参阅<u>第 280 页的『"日志文件状</u> <u>态"属性组』</u>。

日志文件数据源的"特性信息"页面不仅包含适用于所有数据源的字段,还在记录字段信息区域中包含 一些附加字段。

记录字段信息字段包括:

下一个字段

显示使用来自属性组的定界符(或来自 Advanced 对话框的此特性的特殊定界符)解析之后的下一个字段。

记录的其余部分

显示在解析先前的特性之后记录的其余部分。除了可能存在的日志文件名或日志文件标签外,该特性是最后一个特性。

整个记录

显示整条记录,除了可能存在的日志文件名或日志文件标签外,这可能是唯一的特性。

日志文件名

显示日志文件的名称。

日志文件标签

在高级面板上显示指定给文件的标签。

注: 仅当需要派生特性, 而不需要直接来自日志文件的特性时使用派生特性详细信息选项卡。

9. 单击记录字段信息区域中的高级,以显示"高级日志文件特性信息"页面。

a) 在特性过滤器部分中,指定包括或排除数据的条件。

对特性进行过滤可以减少处理的数据量,从而提高解决方案的性能。单击以下一个或多个特性过滤器:

·**包括**:指示特性过滤器集是一个接受过滤器,即表示如果过滤成功,记录将通过过滤器,并将其输出。

- ·**排除**:指示特性过滤器集是一个拒绝过滤器,即表示如果特性过滤成功,那么将拒绝记录,并且不 会输出此记录。
- ·与所有过滤器匹配:指示定义到过滤器的所有过滤器必须都与特性记录匹配,过滤才能成功。
- ·与任何过滤器匹配:指示如果定义到过滤器的任何过滤器与特性记录匹配,即过滤成功。
- b) 使用添加、编辑和移除为特性过滤器集定义各个过滤器。
- c) 要添加过滤器, 请执行以下步骤:
  - 1) 单击添加,并按如下所示填写"添加过滤器"窗口中的选项:

a) 过滤条件部分定义过滤器的基本特征,包括以下属性:

- ·开始偏移量: 定义特性字符串中比较的起始位置。
- ·比较字符串:定义作为特性定义依据的模式字符串。

输入代理程序用来过滤从文件中读取的数据的字符串、模式或正则表达式。与过滤模式匹配 的记录将从返回到监视环境的记录中移除,或者它们是唯一返回的记录。结果取决于您选择 的是 Inclusive 还是 Exclusive 过滤器。

·**匹配整个值**:检查特性字符串中与比较字符串完全一致的匹配项。从起始偏移位置开始查找。

·**匹配值的任何部分**:在特性字符串中的任意位置检查是否存在比较字符串。从起始偏移位置 开始查找。

b) 比较字符串是正则表达式:指示比较字符串是可以针对特性字符串应用的正则表达式模式。

通过使用 International Components for Unicode (ICU) 库提供正则表达式过滤支持,以确定 检查的特性值是否与指定模式匹配。

要有效使用正则表达式支持,必须熟悉 ICU 实现正则表达式的具体方法。此实现方法与 Per1、grep、sed 和 Java 正则表达式中的正则表达式支持的实现方法以及其他实现方法不
同。请参阅 <u>第 309 页的『附录 F ICU 正则表达式』</u>以了解创建正则表达式过滤器的相关准则。

- c) 定义覆盖过滤器:指示您要提供更具体的过滤器比较以覆盖先前定义的基本特征。此额外比较字符串用于逆转过滤结果。如果是包括类型的过滤器,那么覆盖会充当过滤表达式的排除限定词。如果是排除类型的过滤器,那么覆盖会充当过滤表达式的包含限定词。(有关包括和排除的更多信息,请参阅步骤 <u>第 92 页的『9』</u>以及随附示例)。覆盖过滤器具有以下属性:
  - ·开始偏移量: 定义特性字符串中比较的起始位置。
  - ·比较字符串:定义作为特性匹配依据的模式字符串。

输入代理程序用来过滤从文件中读取的数据的正则表达式。与过滤模式匹配的记录将从返回 到监视环境的记录中移除,或者它们是唯一返回的记录。结果取决于您选择的是 Inclusive 还是 Exclusive 过滤器。

- d) **替换值**可用于将原始特性字符串变更为新值。请参阅 <u>第 309 页的『附录 F ICU 正则表达式』</u> 以了解有关可使用的特殊字符的更多详细信息。
- e) 替换第一个实例将比较字符串匹配的第一个匹配项替换为新文本。
- f) 替换所有实例将比较字符串匹配的所有匹配项替换为新文本。

2) 单击确定。

🐵 Add Filter 🛛 🔀
Add Filter
Enter the information needed for a new attribute filter
Filter criteria
Starting offset 0
Comparison string
^([a-z]*) is ([a-z]*) as ([0-9]*)\$
O Match entire value
Match any part of value
The comparison string is a regular expression
Define an override filter
Starting offset
Comparison string
Replacement value
\$3 is not as \$2 as \$1
Replace first occurrence
Replace all occurrences
OK Cancel

图 8. "添加过滤器"示例 1

如果特性字符串是 abc is easy as 123, 那么在 Tivoli Enterprise Portal 或 IBM Cloud Application Performance Management 控制台中显示的已被替换的字符串是 123 is not as easy as abc。

🐵 Add Filter 🛛 🔀
Add Filter
Enter the information needed for a new attribute filter
Flicer criceria
Starting offset 0
Comparison string
Error
O Match entire value
<ul> <li>Match any part of value</li> </ul>
The comparison string is a regular expression
Define an override filter
Starting offset 0
Comparison string
No Errors Found
Replacement value
Replace first occurrence
Replace all occurrences
OK     Cancel

## 图 9. "添加过滤器"示例 2

如果特性字符串是 Unrecoverable Error reading from disk,并且使用**包括**类型的过滤器,那么该特性将显示在 Tivoli Enterprise Portal 或 IBM Cloud Application Performance Management 控制台中。如果特性字符串是 No Errors Found during weekly backup,并且使用**包括**类型的过滤器,那么不会显示此特性。

- d) 在 "高级日志文件特性信息"页面的字段标识部分中,指定如何仅针对这一个特性覆盖属性组字段 分隔符。单击某个特性过滤器,并填写该选项的必填字段:
  - ·字符数: 输入字符数限制。

·制表符分隔符:指定使用制表符分隔符。

·分隔符文本: 输入要使用的分隔符文本。

·开始及结束文本:请同时输入开始文本和结束文本。

e) 在"高级日志文件特性信息"页面的摘要部分中,单击将特性包括在摘要属性组中复选框以将特性 添加到摘要属性组。

当用户开启了日志特性摘要时将生成此属性组。

f) 单击确定。

**10.** 如果在步骤 <u>第 91 页的『7』</u>中使用了测试功能,那么将显示"选择键特性"页面。在"选择键属性" 页面上,选择键属性或指示此数据源仅生成一个数据行。

有关更多信息,请参阅<u>第15页的『选择键属性』</u>。

- 11. 执行下列其中一个步骤:
  - ·如果您正在使用"新建代理程序"向导,请单击下一步。
  - · 单击完成,以保存该数据源并打开 Agent Editor。

**注:** 在缺省的 Tivoli Monitoring 最低版本 (6.2.1) 或更高版本中向代理程序添加了日志文件属性组时,将 包含日志文件状态属性组。有关日志文件状态属性组的更多信息,请参阅<u>第 280 页的『"日志文件状</u> <u>态"属性组』</u>。

# 日志文件解析和分隔符

您可以更改在日志文件记录中用于分隔一个或多个属性的缺省分隔符。

在创建日志文件属性组时,缺省情况下会分配一个分隔符。缺省分隔符是制表符。代理程序使用该分隔符对数据行中每个属性的数据进行解析和定界。您可以将缺省属性分隔符更改为:

- ·固定数量的字符
- ·空格
- ·其他一个或多个字符
- ·特定的开始和结束文本
- ·XML 元素。

您可以通过以下方法更改用于组中所有属性的缺省分隔符:

- 1. 在创建属性组时,在"日志文件信息"页面中更改。
- 2. 创建属性组之后,可通过打开 Agent Editor > 数据源选项卡,选择属性组,然后在"字段标识"区域中选择分隔符。

您还可以选择为一个或多个个别属性指定特定分隔符。您可以将个别属性的特定分隔符指定为使用:

- ·固定数量的字符。
- ·制表符分隔符
- ·空格分隔符
- ·其他一个或多个字符
- ·特定的开始和结束文本。

您可以通过以下方法更改用于个别属性的分隔符:

1. 创建属性时,在"属性信息"页面上选择高级进行更改。

2. 通过打开 Agent Editor > 数据源选项卡,选择属性,并在"日志文件属性信息"选项卡上选择高级。

### 示例 1 - 简单日志文件输出

某些日志文件记录具有明确的常规分隔符,例如:

one,two,three

此处的","字符就是明确的常规分隔符,位于这一行的三段数据之间。在此情况下,选择**分隔符文本**,并 指定","作为属性组的缺省分隔符。无需更改或定义其他分隔符。

以下输出显示了如何为日志文件定义此分隔符,此日志文件包含本示例之前显示的数据行:

Results							
Attribute_2	Attribute_3						
two	three						
	n attributes Attribute_2 two	n attributes Attribute_2 Attribute_3 two three					

图 10. 代理程序解析简单日志文件数据行时的示例属性值输出。

# 示例 2-复杂日志文件输出

有些日志文件可能包含使用非常规或会变化的分隔符的数据行,例如:

one,two,three,[four]12:42,five

在此示例中,可以用于为属性定义指定分隔符的过程如下:

- 1. 在上一个示例中, 将缺省分隔符设置为","。此分隔符用于所有属性, 除非您使用特定分隔符将其覆盖。在此示例中, 仍然使用缺省分隔符","来分隔此行中的前三个属性是正确的。
- 2. 对于第四个属性, 假设"["和"]"之间的字符串是您要抽取的值。在此情况下, 在定义第四个属性时, 为分隔符类型**开始和结束文本**指定开始和结束文本值"["和"]"。
- 3. 对于第五个属性, 假设您要抽取"]"和":"字符之间的值。在此情况下, 在定义第五个属性时, 将分 隔符类型**分隔符文本**设置为":"。
- 4. 对于第六个属性,又可以使用缺省属性组分隔符","。
- 5. 对于第七个属性,无需指定分隔符,因为这是最后一个属性。

以下输出显示了如何为日志文件定义这些分隔符,此日志文件包含本示例之前显示的数据行:

Show hidden attributes								
A	ttribute_1	Attribute_2	Attribute_3	Attribute_4	Attribute_5	Attribute_6	Attribute_7	
0	ne	two	three	four	12	42	five	

图 11. 代理程序解析复杂日志文件数据行时的示例属性值输出。

第 88 页的『监视日志文件』步骤 第 89 页的『5』 中描述了如何定义属性分隔符的过程。

# 测试日志文件属性组

您可以使用 Agent Builder 测试所创建的日志文件数据集(属性组)。如果没有为组定义任何属性,测试过 程将自动进行定义。

### 开始之前

如果已为该数据集定义了任何属性,并想要在测试期间自动定义属性,请使用代理程序编辑器从数据集中移除所有现有的属性。有关指示信息,请参阅<u>第34页的『移除属性』</u>。

# 过程

1. 可以通过以下方法启动测试过程:

· 代理程序创建期间, 在"日志文件信息"页面上单击测试日志文件设置。

 · 在创建代理程序之后,在 Agent Editor 的"数据源定义"页面上选择属性组,然后单击测试日志文件 设置。有关 Agent Editor 的更多信息,请参阅<u>第 17 页的『第 4 章 使用 Agent Editor 修改代理程</u> 序』。

在上述两个步骤中的某个步骤中单击**测试日志文件设置**之后,将打开"**解析日志**"窗口。

- 2. 选择要测试的日志数据的源:
  - · 使用属性组测试: 使用数据源中指定的文件名和位置。缺省情况下,数据源仅处理启动测试过程后添加到日志文件的信息。如果日志文件实时更新,您可以使用此选项。
  - 指定样本文件:提供样本日志文件。通过此设置,测试过程将解析日志文件的整个内容。通过此选项,您可以根据现有样本测试数据源并立即为其创建属性。在日志文件名字段中指定文件的路径和名称,或者使用浏览按钮来选择文件。
- 可选: 在开始测试之前,您可以设置环境变量和配置属性。
   有关更多信息,请参阅第 191 页的『属性组测试』。

## 4. 单击启动代理程序。

此时会打开一个窗口,指示代理程序正在启动。当代理程序启动时,它会监视已配置的日志文件中是否 有新记录

5. 要测试代理程序数据收集,请在受监视日志文件中生成新记录。

在将新记录添加到日志文件时,代理程序将根据其配置解析这些记录,并更新其高速缓存中对应的属性 值。

6. 要模拟来自 Tivoli Enterprise Portal 或 SOAP 的代理程序数据请求,请单击收集数据。

"解析日志"窗口将收集和显示自代理程序上次启动以来高速缓存中的任何新属性值。<u>第97页的图12</u>中显示了示例数据收集

🖻 Parse Log	×
Parse Log	
Select a sample log file to see how it will be parsed.	
C Use attribute group settings	
Specify a sample file	
Log file name C:\Users\mtruss\idwb.rolling.log	Browse
Start Agent Collect Data Stop Agent Check Results Set Environ	ment Configuration
Results	
V Show hidden attributes	
Date	<b>▲</b>
2012-01-28 15:18:45 [DEBUG] [main] IDV/BInfo - ID Workbench version 4.3.1 is installed at "C:\Program Files (x86)]BM\IDV/B"	
2012-01-28 15:18:45 [DEBUG] [main] IDWBComponentInfo - Preparing component information for component ACRO	
2012-01-28 15:18:45 [DEBUG] [main] ID/WBC/mponentinino - Prepandig component information for component EPIC	
2012-01-20 15:10-75 [DEB06] [Imain] id/vbcom/poinertanio - Prepang Component mormation to component back	
2012-01-20 15:10-10 [DED00] [main] http://dominantecommonantecom experimentecommonantecommonantecommonantecommonantecommonantecommonantecommonantecommonantecommonantecommonantecommonantecomm	
2012-01-28 15:18:5 [DFBIIG] [main] ToolCount - Default tool version will be (4.3.1)	
2012-01-28 15:18:45 [DEBUG] [main] idwb - com.ibm.idwb.common.toolcount.ToolCount logging initialized	
2012-01-28 15:18:45 [DEBUG] [main] IDWBInfo - ISDEVELOPMENT false	
2012-01-28 15:18:45 [DEBUG] [main] IDWBInfo - ISPRERELEASE false	-
	Þ
3	OK Cancel

图 12. 显示已解析的日志文件属性值的 "解析日志" 窗口

7. 可选: 如果返回的数据未如预期, 请单击检查结果。

"数据收集状态"窗口将会打开,并显示有关该数据的更多信息。<u>第 239 页的『Performance Object</u> Status 节点』中描述了 Data Collection Status 窗口收集和显示的数据

- 8. 通过单击停止代理程序可停止代理程序。
- 9. 单击确定或取消以退出"解析日志"窗口。单击确定将保存您所作的全部更改。

# 相关概念

<u>第191页的『在Agent Builder 中测试代理程序』</u>

# 监视 AIX 二进制日志

您可以定义数据源以通过 errpt 命令监视 AIX 二进制错误日志。您还可以对其进行配置以过滤和汇总数 据。产生的事件放置在数据集中。

### 关于此任务

日志监视功能支持通过 errpt 命令监视 AIX 二进制错误日志。errpt 命令根据错误日志中的条目生成错误 报告。它提供了用于选择与特定条件相匹配的错误的标志。对通过 errpt 命令监视 AIX 二进制错误日志的 此支持,是以 Tivoli Monitoring UNIX Logs Agent (产品代码为 kul 或 ul)中的同一功能支持为基础而设 计。

向 Agent Builder 提供 **errpt** 命令字符串后,它将处理由于运行此命令而发生的事件。Agent Builder 将对 此命令实施 Monitoring Agent for UNIX Logs 所实施的约束。尤其是,必须使用 -c(并发方式)选项以使此 命令持续运行,并且不得使用 -t 选项或下列将产生详细输出的选项: -a、-A或 -g。

监视 AIX **errpt** 命令的 Agent Builder 代理程序将自动包括 Monitoring Agent for UNIX Logs 所包括的信息。有关 AIX 二进制错误日志的属性组的更多信息,请参阅<u>第 252 页的『"AIX 二进制日志"属性组</u>。

## 过程

- 1. 在"代理程序初始数据源"页面或"数据源位置"页面中,单击监视数据类别区域中的记录的数据。
- 2. 在数据源区域中,单击 AIX 二进制日志。
- 3. 单击下一步。
- 4. 在"二进制日志信息"页面中, 输入 errpt 命令。

缺省值为:

errpt -c -smmddhhmmyy

启动时,代理程序将搜索"mmddhhmmyy"字符串并将其替换为实际日期和时间。将只替换此字符串的 第一个实例。

您可以提供自己的 errpt 命令,但 Agent Builder 将对此命令实施 Monitoring Agent for UNIX Logs 所实施的约束。尤其是,必须使用-c(并发方式)选项以使此命令持续运行,并且不得使用-t选项或下列将产生详细输出的选项: -a、-A或-g。

- 5. (可选)单击**高级**以选择事件的过滤及摘要选项。有关更多信息,请参阅<u>第 219 页的『控制重复事件』</u>。
- 6. 执行下列其中一个步骤:
  - ·如果您正在使用"代理程序"向导,请单击下一步。
  - · 单击完成,以保存该数据源并打开 Agent Editor。

#### 相关参考

第 252 页的『"AIX 二进制日志"属性组』

"AIX 二进制日志"属性组显示来自 AIX 二进制日志的事件,这些事件由提供的 errpt 命令字符串选择。

# 监视 Windows 事件日志

您可以定义数据源以从 Windows 事件日志收集数据。您可以对其进行配置以过滤数据。产生的事件放置在"事件日志"数据集中。

## 关于此任务

您可以使用事件类型、源或标识从 Windows 事件日志收集数据。您可以使用这些参数来过滤 Windows 系统 收集的日志事件。代理程序将受监视事件日志中的每个新事件与指定的过滤器进行比较。如果事件与过滤器 中指定的某个事件类型、事件源以及事件标识相匹配,那么该事件将通过。

例如,如果事件日志过滤器针对的是应用程序日志,请指定错误作为事件类型。此选项与记录到应用程序日志并且事件类型值为 error 的所有事件匹配。如果添加 Diskeeper 和 Symantec AntiVirus 事件源,那么 代理程序将与来自其中任一源的所有错误事件匹配。可添加特定事件标识来进一步优化过滤器。事件类型、事件源和事件标识之间不存在直接关联。如果其中某个值各自与事件相匹配,事件即匹配。

缺省情况下,仅处理代理程序启动之后生成的事件。但是,您可以允许代理程序在重新启动时处理其关闭期间生成的日志事件。有关使代理程序能够处理在代理程序关闭期间生成的事件的更多信息,请参阅步骤<u>第</u>99页的『6』。

## 过程

- 1. 在"代理程序初始数据源"页面或"数据源位置"页面中,单击监视数据类别区域中的记录的数据。
- 2. 在数据源区域中,单击 Windows 事件日志。
- 3. 单击下一步。
- 4. 在"Windows 事件日志"页面上,从 Windows 事件日志名称列表中选择其中一个日志的名称,或者 为事件日志输入名称。

该列表由当前系统上的日志集合构成,例如:

应用程序 安全性 系统

- 5. 在"Windows 事件日志"页面中,指定您是要使用以下其中一种还是多种机制来过滤结果:
  - · 第100页的『按事件类型过滤』
  - · 第 100 页的『按事件源过滤』
  - · 第 101 页的『按事件标识过滤』

注:必须至少选择其中一个过滤条件。

6. 要处理在代理程序关闭期间生成的日志事件,请在代理程序重新启动时,单击"Windows 事件日志" 页面上的 脱机事件设置。

"Windows 事件日志书签设置"窗口随即打开。

7. 选择下列其中一个书签选项:

注:下列选项将应用于所有受监视的 Windows 事件日志。

- · 不收集任何脱机事件: 不处理在代理程序关闭期间生成的事件。此选项为缺省选项。
- · 收集所有脱机对象:处理在代理程序关闭期间生成的所有事件。
- · 指定定制收集设置:您可以输入值以根据时间值和/或事件数量来调整旧事件的处理速度。通过使用此选项,可确保在代理程序启动时,监视环境不会因为事件过多而超负荷。

例如,如果在**要收集的最大事件数**字段中输入了100,在根据时间间隔限制收集(秒)字段中输入了30。那么处理的事件数量为代理程序启动之前生成的最后100个事件,或者是代理程序启动30秒内生成的任何事件。其结果取决于首个匹配的变量。

如果输入了要收集的最大事件数的值,那么将添加 CDP\_DP\_EVENT\_LOG\_MAX\_BACKLOG\_EVENTS 环境变量。在输入值以根据时间间隔限制收集时,会添加

CDP\_DP\_EVENT\_LOG\_MAX\_BACKLOG\_TIME 环境变量。添加任意一个或同时添加这两个变量时,会 创建 eventlogname\_productcode\_instancename\_subnodename.rst 文件,其中包含为事 件日志处理的最后一个事件记录。此文件位于 %CANDLE\_HOME%\tmaitm6\logs 目录中,并在重 新启动代理程序以处理代理程序关闭期间生成的旧事件时使用。

8. 如果要为数据源设置全局选项,请单击"Windows 事件日志"页面上的全局选项。

"全局 Windows 数据源选项"窗口随即打开。

9. 如果要包括此选项,请选中包括远程 Windows 配置属性复选框,并单击确定。

有关 Windows 数据源的 Windows 远程连接配置的信息,请参阅<u>第 177 页的『配置 Windows 远程连</u>接』。

- 10. 在指定过滤器并单击确定之后,请在"Windows 事件日志"页面上执行下列其中一个步骤:
  - ·如果您正在使用"代理程序"向导,请单击下一步。
  - · 单击完成,以保存该数据源并打开 Agent Editor。新 Windows 事件日志的名称将显示在 Agent Editor 的 "数据源定义"页面上。

# 下一步做什么

有关 Windows 事件日志数据源的 Windows 远程连接配置的信息,请参阅<u>第 177 页的『配置 Windows 远程</u> 连接』。

# 按事件类型过滤

按事件类型过滤 Windows 事件日志结果

### 过程

1. 在"Windows 事件日志"页面中,选择按事件类型过滤。

2. 选择以下一个或多个事件类型:

- ・参考
- ・ 警告
- ・错误
- · 成功审计
- · 失败审计
- 3. 单击完成以完成操作。

# 按事件源过滤

按事件源过滤 Windows 事件日志结果

# 过程

- 1. 选择按事件源过滤,并在"Windows 事件日志"页面的事件源区域中单击添加。 "事件源"窗口随即打开。
- 2. 选择执行下列操作之一。
  - · 输入事件源名称, 然后单击确定。
  - · 单击浏览以查找并从列表中选择事件源, 然后单击确定。

您选择的名称会显示在"事件源"窗口中。

注:

- a. 要对事件源列表进行排序,请单击列标题。
- b. 要刷新窗口中的信息, 单击刷新图标。

c. 要搜索特定事件源,单击搜索(望远镜)图标。

3. 单击确定后,您将在"Windows 事件日志"窗口的"事件源"列表中看到这个新增的事件源过滤器。

#### 按事件标识过滤

对于 Windows 事件日志数据源,您可以按事件标识过滤事件。

#### 关于此任务

要按事件标识进行过滤,请使用以下过程:

#### 过程

- 1. 选择按事件标识过滤,并在"Windows 事件日志"窗口的事件标识区域中单击添加。 这将显示"事件标识"窗口。
- 2. 如果您要监视来自应用程序的特定事件,请在应用程序定义事件时指定事件的编号。输入整数以作为事件标识,然后单击确定。

此时,该新事件过滤器的数字标识显示在"Windows 事件日志"的"事件标识"列表中。

注:必须逐个定义每个事件标识。

- 3. 如果要修改某个 Windows 事件日志,请将其选中,然后单击编辑。
- 4. 如果要删除某个 Windows 事件日志,请将其选中,然后单击移除。
- 5. 您可以继续向列表添加事件日志,也可以单击完成。

# 监视命令返回码

您可以定义数据源以使用命令返回码来监视应用程序或系统。代理程序运行命令,收集返回码,并将结果添 加到"可用性"数据集。

#### 关于此任务

用户创建的脚本、可执行文件、查询或系统命令能够返回代码。命令返回码是用于确定应用程序或受监视的系统是否可用的特定于应用程序的机制。代理程序将运行指定的命令并通过检查返回码确定应用程序或受监视的系统的状态。

命令必须对每种描述性状态都显示一个唯一返回码。命令还必须对其中每个返回码都定义要由代理程序使用的消息。命令可以在用户创建的脚本、可执行文件、查询或系统命令内使用环境和配置变量。以命令行方式调用命令时,该命令不能使用环境或配置变量,仅下列变量除外: AGENT\_BIN\_DIR、AGENT\_ETC\_DIR、AGENT\_LIB\_DIR、CANDLE\_HOME和 CANDLEHOME。

#### 过程

1. 在"代理程序初始数据源"页面或"数据源位置"页面中,选择监视数据类别区域中的命令或脚本。

- 2. 在数据源区域中,单击命令返回码。
- 3. 单击下一步。
- 4. 在"命令返回码"页面的命令返回码信息区域中, 输入显示名。
- 5. 使用以下子步骤来定义和描述您希望命令返回码使用的命令行。

**注:** 请为代理程序支持的每一款操作系统定义一个命令。这些命令可以共享,但所有命令对应的操作系统的总集合必须与代理程序支持的操作系统集合相同。

- a) 在"命令返回码"窗口的命令区域中单击添加以打开"命令信息"窗口。
- b) 输入命令行, 并从"命令信息"窗口的操作系统区域中的列表中选择操作系统。
  - 注:

- 1) 对于 Windows 命令,必须输入命令的全名。例如, command\_to\_run.bat, 而不仅仅是 command\_to\_run。
- 将名称加上引号,以免命令解释器对其进行语法分析。例如,请输入"this is a test.bat" argument,而不要输入this is a test.bat argument。
- 3) 您可以单击命令, 然后单击编辑对其进行修改, 或者单击删除将其删除。
- c) 在"命令信息"窗口的 Return Codes 区域中, 单击添加。
- d) 在"返回码定义"窗口中, 从显示的列表中选择返回码类型。

您可以指定下列状态以测试返回码:

- ALREADY\_RUNNING
- · DEPENDENT\_NOT\_RUNNING
- $\cdot \text{GENERAL}_\text{ERROR}$
- ·NOT\_RUNNING

· 0K

- PREREQ\_NOT\_RUNNING
- ・WARNING
- e)为选择的返回码类型输入数字值。

返回码的值是一个整数,用于指定针对命令返回码定义的返回码。为了能够在操作系统之间进行移 植,请使用 0 - 255 范围内的返回码值。对于仅在 Windows 上运行的命令,返回码值可以从 -2147483648 - 2147483647。

f) 请为每个返回码定义一条消息,以便可以同时显示消息和代码。单击浏览以设置消息文本。 消息窗口列出代理程序中定义的消息。"消息"(列表)窗口将打开。

注:

- 1) 通过在消息文本列表中进行选择,即可选择先前输入的文本,而无需单击**浏览**。然后,转到步骤 <u>5k</u>。
- 2) 在定义消息之前,列表保留为空。您可以使用**编辑**来变更已定义的消息,并可以使用**移除**来删除一条或多条已定义的消息。
- g) 在"消息"(列表) 窗口中,单击添加。

"消息定义"窗口随即打开。

注:系统会为您自动生成消息标识。

- h) 在消息文本字段中输入一些文本, 用于描述新消息的含义。
- i) 单击**确定**。

"消息"(列表)窗口将打开,并显示新消息。

j) 要验证该消息并使其永久化,请从列表中选择该消息,然后单击确定。

新的返回码类型、值和文本将显示在"返回码定义"窗口中。

- k) 如果您希望此返回码可用于此命令返回码的其他操作系统上的其他命令,请选中**全局返回码适用于所** 有命令。如果您想要让此返回码仅可用于此命令,请将本地返回码仅适用于此命令保留为选中状态。
- l) 在"返回码定义"窗口中, 单击窗口。
- m) 在离开"**命令信息**"窗口之前,请至少定义两个返回码。一个返回码用于指示没有可用性问题,另一个用于指示是否发生问题。如果要添加另一个返回码,请返回到步骤 <u>c</u>。
- n) 可选:如果您想选择一个或多个脚本或可执行文件以供代理程序运行,请在"命令信息"窗口的命令 文件区域中单击添加。

一个或多个文件将复制到代理程序的项目文件夹的 scripts/operating system 之下,其中 operating system 是一个变量,它依赖于您在"命令信息"窗口的操作系统区域中选择的内容。这些

文件也与代理程序一起打包和分发。要编辑现有命令文件的定义,或者编辑已复制至项目的原始命令 文件,请选中该文件,然后单击编辑。请参阅<u>第104页的『编辑命令文件定义』</u>。

o) 在"命令信息"窗口中, 单击确定。

**注:** 命令文件表是定义要包含在代理程序包中的所有外部文件的位置。这些文件将复制到项目目录中,并随代理程序一起打包,以进行分发。

- 6. 如果存在其他尚未定义的返回码,请定义并描述命令返回码可以使用的全局返回码。
  - a) 在"命令返回码"页面的"全局返回码"区域中,单击添加。

注:此处定义的返回码是全局返回码。这表示这些返回码适用于所有针对命令返回码定义的命令。 (这些返回码不能在命令返回码之间共享。)另外,您可以在输入命令信息时定义返回码。在这里定 义的返回码可以是全局返回码或局部返回码。本地返回码仅适用于此特定命令。如果具有跨所有操作 系统都相同的返回码,则该层次结构很有用。(例如,返回码 0 表示一切正常。您可以在全局级别对 它进行定义,然后所有已定义命令都按这种方式来解释 0。)如果其他所有操作系统都未返回 5,那 么可以仅为 Windows 命令定义返回码 5。如果在本地命令级别定义的返回码与已在全局级别定义的返 回码相同,那么使用该命令级别。您可以使用此方法来覆盖特定操作系统上的返回码。例如,即使返 回码 2 在所有 UNIX 操作系统上都表示某个含义,但是在 Windows 上,它表示另一种含义。您可以在 全局级别按 UNIX 操作系统的需要定义返回码 2。然后,在 Windows 的命令中,可以针对 Windows 上的含义重新定义返回码 2。

b) 在"返回码定义"窗口中,从显示的列表中选择返回码类型。

您可以指定下列状态以测试返回码:

- ALREADY\_RUNNING
- DEPENDENT\_NOT\_RUNNING
- •GENERAL\_ERROR
- •NOT\_RUNNING
- · 0K
- PREREQ\_NOT\_RUNNING
- · WARNING
- c)为选择的返回码类型输入数字值。返回码的值是一个整数,用于指定针对命令返回码定义的返回码。
- d) 单击**浏览**以设置消息文本及其相关含义。必须为每个返回码定义一条消息,以便可以同时显示消息和 代码。

"消息"窗口列出代理程序中定义的消息。

注:

- 1) 在定义消息之前,列表保留为空。您可以使用编辑来变更已定义的消息,并可以使用移除来删除一条或多条已定义的消息。
- 2) 通过在**消息文本**列表中进行选择,即可选择先前输入的文本,而无需单击**浏览**。然后,转到步骤 <u>6h</u>。
- e) 在"消息"(列表)窗口中,单击添加以查看"消息定义"窗口,您可以在该窗口中输入文本以描述 新消息的含义。
- f) 单击确定。
- g) "**消息**" (列表) 窗口将打开, 并显示新消息。要验证该消息并使其永久化, 请从列表中选择该消息, 然后单击**确定**。
- h) 当新的文本、类型和值显示在"返回码定义"窗口中之后, 单击确定。
- i) 在"命令返回码"页面上,当您完成针对所有受支持操作系统定义返回码及命令时,执行下列其中一个步骤:
  - ·如果您正在使用"新建代理程序"向导,请单击**下一步**,或者单击**完成**以保存该数据源并打开 Agent Editor。
  - ·如果您正在使用"新建代理程序组件"向导,请单击完成以返回到 Agent Editor。

#### 下一步做什么

如果想要在 IBM Cloud Application Performance Management 的摘要仪表板中使用来自此数据源的数据, 必须创建基于"可用性"数据集过滤的数据集(属性组),并将其配置为提供单行。使用"名称"字段来选 择用于您的进程的行。

在新的过滤属性组中,选择"状态"字段并为其指定严重性值。

有关指示信息,请参阅:

- ·第155页的『创建过滤属性组』
- ·第38页的『指定用作状态指示符的特性的严重性』
- ·第 187 页的『第 12 章 为 Cloud APM 准备代理程序』

# 编辑命令文件定义

您可以对导入到项目中的命令文件进行更改,也可以将针对现有命令文件的更改导入到项目中。

# 过程

1. 在"命令信息"窗口的"命令文件"区域中选择文件。

2. 单击编辑以打开"导入命令文件"窗口。

从"**导入命令文件**"窗口中,您可以了解该命令文件的状态。您还可以更改初始源文件的位置,并将源 文件重新复制到代理程序中。

- 3. 选择下列某个步骤:
  - · 单击确定, 以将文件复制安排在下一次保存代理程序时执行。
  - · 单击**立即复制**以复制该文件,而不首先保存代理程序。

注:从"新建代理程序"向导中访问"导入命令文件"窗口时,立即复制选项不可用。

#### 文件拆分与合并

您可以使用"拆分与合并"功能将文件移入和移出代理程序中特定于操作系统的文件夹。

文件首次添加到代理程序时,将在 scripts/all\_windows 文件夹、scripts/all\_unix 文件夹或 scripts/common 文件夹中添加一个副本。如果在 Windows 和 UNIX 上都使用该文件,那么将使用 scripts/common 文件夹。

要在不同操作系统上放置此文件的不同副本(例如,二进制可执行文件),请单击**编辑**,然后单击**分隔**。系 统会将该文件从公共文件夹中移除,并将其复制到特定于操作系统的文件夹中。然后,可以将该文件的各个 副本替换为适合于特定操作系统的副本。

注: Java 资源文件必须保留在 scripts/common 文件夹中。您无法通过单击分隔来针对个别操作系统建立 Java 资源文件的单独副本。

如果您已将文件分隔到操作系统文件夹中,那么可以使用**合并**将其移回到公共文件夹。如果您是在不支持公 共文件夹的 Agent Builder 版本中创建代理程序,请使用**合并**将这些文件移回到公共文件夹。如果该文件的 任何副本与其他副本不同,系统会提示您选择用作公共文件的文件。将废弃所有其他副本。

# 监视脚本的输出

您可以定义数据源以从脚本或外部程序收集数据。当通过标准管理界面无法使用应用程序数据或需要在单个 行中提供多行数据摘要时,使用该数据源。代理程序运行脚本并收集其输出。脚本输出中的每一行都会解析 为所产生数据集中的一行。

可以从本地或远程系统收集数据。脚本或程序输出只能包含属性组中每个属性的值。要返回多行数据,必须 使用换行符来分隔各行数据。每个数据行中的属性都由您定义的分隔符分隔。有关分隔符的更多信息,请参 阅第 105 页的『脚本解析和分隔符』

此命令可以在用户创建的脚本、可执行文件、查询或系统命令中使用环境变量和配置变量。以命令行方式调用此命令时,此命令无法使用除下列变量以外的环境变量或配置变量:AGENT\_BIN\_DIR、AGENT\_LIB\_DIR、CANDLE\_HOME和 CANDLEHOME。

代理程序将监视以代理程序运行环境的语言环境和代码页写入的脚本输出。

## 从远程系统收集脚本数据

为了从远程系统收集脚本或程序数据, Agent Builder 使用 Secure Shell (SSH)

为了从远程系统收集数据, Agent Builder 会创建一个 Secure Shell (SSH) 会话,并在远程系统上启动脚本或 外部程序。代理程序建立并登录到 SSH 会话。然后,代理程序将脚本上载到远程系统,启动脚本或外部程 序,并检索输出。可以对代理程序进行配置,以便该会话保持打开状态或者为每个调用重新建立该会话。如 果该会话保持打开状态,那么可以对每个调用重复使用或上载脚本。在缺省情况下,将使用单个 SSH 会话并 对每个调用重复使用脚本。

Agent Builder 仅支持将 SSH 协议 V2 与 Rivest、Shamir 和 Adleman (RSA) 或 Digital Signature Algorithm (DSA) 密钥一起使用。该代理程序通过用户名和密码或公用密钥进行认证。公用密钥的生成和分发是一项必须在代理程序和 Agent Builder 外部执行的管理任务。

要在远程系统上针对启用了 Secure Shell (SSH) 的脚本数据提供程序运行所编写的"执行操作"命令,请参阅 <u>第 325 页的『SSHEXEC 操作』</u>。

**限制:** 如果您的代理程序是使用 Agent Builder V6.3 之前的版本构建的,并且其脚本数据提供程序使用 SSH,那么在与 IBM Tivoli Monitoring V6.3 或更高版本一起运行时,该提供者将失败。要解决此问题,请使 用 Agent Builder 的当前版本重新构建代理程序。

存在此限制是因为 IBM Tivoli Monitoring V6.3 使用较新版本的全局安全工具箱 (GSKit) API。您必须使用 Agent Builder 6.3 或更高版本来重新构建代理程序,才能将其与 IBM Tivoli Monitoring V6.3 或更高版本一起 运行。如果使用 Agent Builder 6.3 构建代理程序,那么它也可以与较低版本的 IBM Tivoli Monitoring 一起运 行。

# 脚本解析和分隔符

您可以为一个或多个属性更改和指定特定的脚本分隔符。

在创建脚本属性组时,缺省情况下将指定单字符文本分隔符。该缺省分隔符是";"。代理程序使用该分隔 符对数据行中每个属性的数据进行解析和定界。您可以更改缺省分隔符以使用其他字符。您还可以为一个或 多个个别属性指定特定分隔符。

您可以为个别属性指定特定分隔符:

- ·使用输出中固定数量的字节。
- ·以定制分隔符分隔相邻属性,该分隔符可以由多个字符组成。
- ·在属性值的开头和结尾用字符串为该值定界。
- ·将其余文本作为属性值返回(无论它是否包含嵌入的分隔符)。

您可以使用一个或多个此类分隔符从数据行中抽取属性值。

#### 示例 1 - 简单脚本输出

有些脚本可以输出带有明确的常规分隔符的数据行,例如:

Row One;1;2 Row Two;3;4 Row Three;5;6

此处的";"字符就是明确的常规分隔符,位于每一行的三段数据之间。在此情况下,使用缺省分隔符即 可,因此无需更改或定义其他分隔符。不难想象使用其他字符作为分隔符的类似脚本输出,如同以下示例。

Row One-1-2 Row Two-3-4 Row Three-5-6

在此示例中, 分隔符从";"字符更改为"-"字符。在此情况下, 在定义属性时, 更改缺省分隔符以使用 "-"字符。

### 示例 2 - 复杂脚本输出

有些脚本可能输出使用非常规或会变化的分隔符的数据行,例如:

Row One;1;2;[option]Hour:MIN;fourtabby The end;4 Row Two;3;4;[required]12:30;fourvery tabby the tail;5 Row Three;5;6;[out]March:12;fourline up the rest of the story;6

在此示例中,可以用于为属性定义指定分隔符的过程如下:

- 1. 最初可使用缺省分隔符";"分隔每个数据行中的前三个属性。在此情况下,将分隔符类型**分隔符文本**设置为";",这样在定义每个属性时,此设置是缺省设置。
- 2. 对于第四个属性, 假设"["和"]"之间的字符串是您要抽取的值。在此情况下, 在定义第四个属性时, 为分隔符类型**开始和结束文本**指定开始和结束文本值"["和"]"。
- 3. 对于第五个属性, 假设您要抽取"]"和":"字符之间的值。在此情况下, 在定义第五个属性时, 将分 隔符类型**分隔符文本**设置为":"。
- 4. 对于第六个属性,又可以使用缺省分隔符";",因此请接受缺省值。
- 5. 对于第七个属性,假设您要抽取接下来的四个字符组成的字符串"four"。在此字符串结尾没有明确的 分隔符。您可以指定一定数量的字符来定义其与下一个属性的分界。您可以分配分隔符类型**字符数**,并 指定长度为4个字符。
- 6. 对于第八个属性,您要抽取字符串 tabby、very tabby 和 line up。在这种情况下,您可以假定所有 这些字符串后面都跟着一个制表符。此时,分配类型为**制表符分隔符**的分隔符。
- 7. 对于第九个属性, 您应该再次使用缺省分隔符类型, 以将剩余文本抽取到此属性。

8. 对于第十个属性,指定记录的其余部分以将数据行的剩余部分分配给此属性。

以下输出显示了如何在脚本中定义这些分隔符,此脚本输出本示例之前显示的数据行:

Results									
✓ Show hidden attributes									
Attribute_1	Attribute_2	Attribute_3	Attribute_4	Attribute_5	Attribute_6	Attribute_7	Attribute_8	Attribute_9	Attribute_10 (Remainder of record)
Row One	1	2	option	Hour	MIN	four	tabby	The end	4
Row Two	3	4	required	12	30	four	very tabby	the tail	5
Row Three	5	6	out	March	12	four	line up	the rest of the story	6

图 13. 代理程序解析复杂脚本输出时的属性值输出示例。

第 106 页的『监视脚本输出的步骤』步骤 第 108 页的『10』 中描述了如何定义属性分隔符的过程。

### 监视脚本输出的步骤

配置代理程序以接收来自脚本数据源的数据。

#### 开始之前

请参阅: <u>第104页的『监视脚本的输出』</u>。

#### 关于此任务

使用以下过程可监视脚本输出:

#### 过程

- 1. 在"代理程序初始数据源"页面或"数据源位置"页面中,选择监视数据类别区域中的命令或脚本选项。
- 2. 在数据源区域中,单击脚本输出。
- 3. 单击下一步。
- 4. 在"命令列表"页面上,单击添加以显示"命令信息"窗口。

**注:** 选中**允许使用 SSH 进行数据收集**复选框可为此属性组启用 SSH。如果此复选框未处于选中状态,此 属性组以本地方式运行。

**注:** 如果有可以在 Agent Builder 运行所在的操作系统上运行的命令,将启用**测试**选项。您可以使用**测试** 来测试所定义的命令。

5. 在 "**命令信息**" 窗口的**命令信息**区域中,在**命令**字段中输入命令名和必需自变量,在**分隔符**字段中输入 分隔符。

注:

a. Windows 中的脚本频繁启动而不在命令行中指定.bat 或.cmd 扩展名。对于远程执行,必须安装 shell 环境,并且必须在脚本数据源命令中指定.bat 或.cmd,这样脚本才能运行。Cygwin 就是一 个可用于 Windows 的 shell 环境示例。Linux、Red Hat 和 AIX。要验证 shell 环境是否存在,请通过 SSH 访问远程主机或登录远程主机并输入以下命令:

PATH=\$PATH:. <command>

如果此命令运行成功,那么表明存在 shell 环境。

b. 将名称加上引号, 以免命令解释器对其进行语法分析。例如, this is a test.bat argument 变成:

"this is a test.bat" argument

c. 环境变量和配置变量可以在用户提供的脚本中使用,但不能作为启动该脚本的命令行的组成部分。 此规则不适用于以下变量:

AGENT\_BIN\_DIR

这是代理程序用于存放二进制文件或脚本的目录

AGENT\_ETC\_DIR

这是代理程序用于存放配置文件的目录

#### AGENT\_LIB\_DIR

这是代理程序用于存放共享库或动态链接库的目录

#### CANDLEHOME

Linux 或 UNIX Tivoli Monitoring 安装目录

#### CANDLE\_HOME

Windows Tivoli Monitoring 安装目录

- d. 如果正在使用 SSH 数据收集选项,将以远程系统用户主目录的相对路径运行命令行。如果您将脚本 或可执行文件上载到远程系统,那么它们将复制到代理程序的环境变量 CDP\_SSH\_TEMP\_DIRECTORY 中指定的位置。缺省位置为远程系统上的用户主目录。在某些系统上,您可能需要使用相对路径 (例如,./Script.sh)定义命令行。
- 6. 在操作系统区域中,选择一个或多个操作系统。使用 SSH 从远程系统收集数据时,"操作系统"是安装 代理程序的系统的属性。而不是远程系统的"操作系统"。在使用 SSH 数据收集功能时,建议选中所有 操作系统复选框。
- 7. 可选: 如果一个或多个用户定义文件是运行命令所必需的, 请单击"命令文件"区域中的**添加**以指定系 统中的文件。

文件复制到代理程序项目文件夹的 scripts/operating system下,其中,operating system 是一 个变量,取决于您在"**命令信息**"窗口中选择的内容。这些文件也与代理程序一起打包和分发。如果要 编辑已添加的命令文件定义,或者更改其内容,请选择该文件并单击编辑。请参阅<u>第104页的『编辑命 令文件定义』</u>。

- 8. 单击确定。"命令列表"页面随即显示。
- 9. 要测试命令,请使用以下步骤:
  - a) 单击测试以打开命令信息并显示"测试命令"窗口。要在远程系统上测试脚本,请从连接名称列表 中选择系统,或者单击添加以添加系统主机名。
  - b) 使用"**测试命令**"窗口来更改命令、缺省分隔符和特性分隔符,以及查看这些更改对返回的数据有 何影响。

1) 在字段中输入命令和分隔符(若还未输入)。

**注:**您可以指定其他分隔符,方法是在创建特性时使用"**特性信息**"窗口,或者使用 Agent Editor 来修改现有特性。有关 Agent Editor 的更多信息,请参阅<u>第 17 页的『第 4 章 使用 Agent Editor</u> 修改代理程序』,有关操作数据源和特性的更多信息,请参阅<u>第 31 页的『第 5 章 编辑数据源和</u> 特性属性』

- 2) 在开始测试之前,您可以设置环境变量和配置属性。有关更多信息,请参阅<u>第191页的『属性组</u>测试』。
- 3) 单击确定, 以返回到"测试设置"窗口。
- 4) 单击启动代理程序。此时将显示一个窗口,表明代理程序正在启动。
- 5) 要模拟来自 Tivoli Enterprise Portal 或 SOAP 的代理程序数据请求,请单击**收集数据**。Agent Builder 会运行您的命令。如果已指定远程系统,请提供用户标识和密码。即使返回码不是 0, Agent Builder 仍将以与代理程序相同的方式解析命令的结果。
- 6) "测试设置"窗口会收集并显示自代理程序上次启动以来,代理程序高速缓存中的全部数据。特性的初始名称是 Attribute\_1 和 Attribute\_2,依此类推;但是,您可以通过单击相应的列标题来修改特性的属性。
- 7) 单击检查结果可查看命令返回码、未解析的数据以及返回的任何错误消息。
- 8) 通过单击停止代理程序可停止代理程序。
- 9) 单击确定, 以返回到"命令信息"窗口。

如果更改了命令或分隔符,那么将更新相应命令来反映这些更改。

如果在创建脚本数据源时打开过此窗口,那么特性已添加到新脚本数据源。

如果该窗口是从现有脚本数据源打开的,那么对特性的任何更改都会应用于该脚本数据源。将添加任何附加特性,但不会移除任何多余的特性。这些选项仅影响从脚本输出解析的特性。不会影响任何派生的特性。如果其中任何特性根据其引用的特性而变为无效,那么可以手动更新或移除派生的特性。显示的是派生特性公式而不是实际的结果值。

注:如果属性组存在,那么请完成以下过程来启动测试

- a. 在 Agent Editor 的"数据源定义"页面上选择属性组。
- b. 从"命令列表"中选择要测试的脚本
- c. 单击测试,并执行步骤 <u>第107页的『9』</u>的过程。
- 10. 如果跳过步骤 (第 107 页的『9』) 中的命令测试,请使用以下步骤:
  - a) 在填写了命令信息的"命令列表"页面上,单击下一步。
  - b) 在"特性信息"页面上,使用<u>第 35 页的表 5</u> 来填写特性名称和类型信息。选择**添加其他特性**以添加 其他特性。
  - c) 在"特性信息"页面上,使用"脚本特性信息"选项卡为此特性选择特定数据分隔符。 缺省情况下已选择标准分隔符;。您可以选择多种其他分隔符,例如字符串、一定数量的字符、制表 符或空格。您也可以选择使用分别用于数据开头和结尾的不同字符串分隔符。最后,还可以选择记 录的其余部分来向特性分配剩余记录。有关脚本解析和分隔符的更多信息,请参阅<u>第105页的『脚</u> 本解析和分隔符』。
- 11. 执行下列其中一个步骤:
  - ·如果您正在使用"代理程序"向导,请单击下一步。
  - ·单击完成,以保存该数据源并打开 Agent Editor。
- 12. 您可以添加特性并提供其相关信息。有关更多信息,请参阅<u>第32页的『创建属性』</u>。
  - 脚本数据源的"**数据源定义**"页面除了包含适用于所有数据源的字段(如<u>第 34 页的『用于定义特性的</u> 字段和选项』所述)以外,还包含以下选项:
  - 命令列表

提供对命令和脚本的访问以在数据收集过程中启动。

添加

允许用户添加可由此属性组启动的命令。

编辑

允许用户编辑现有命令条目。

移除

允许用户删除现有命令条目。

# 测试

允许用户访问此属性组的测试环境。

# 允许使用 SSH 进行数据收集

选中此复选框将对此属性组启用 SSH。如果此复选框未处于选中状态,此属性组以本地方式运行。 有关脚本数据源的 SSH 远程连接配置的信息,请参阅第 179 页的『配置安全 Shell (SSH) 远程连接』。

# 监视来自 Java 数据库连接 (JDBC) 的数据

您可以定义数据源以从 JDBC 数据库接收数据。代理程序运行 SQL 查询以从数据库收集数据。查询返回的每 个列都将成为所产生数据集中的一个特性。

# 关于此任务

JDBC 数据提供程序支持以下数据库服务器:

- ·IBM DB2<sup>®</sup> 9.x 和 8.x
- $\cdot$  Microsoft SQL Server 2008、2005 和 2000
- · Oracle Database 11g 和 10g

Agent Builder 不包含这些数据库的 JDBC 驱动程序。JDBC 驱动程序是与数据库建立 JDBC 连接所必需的一组 JAR 文件,并由供应商提供。为方便起见,下面提供了到可下载的驱动程序的链接:

·IBM Db2: 在数据库服务器安装的 Db2 主安装目录下名为 java 的子目录中包含了 JDBC 驱动程序。

· Microsoft SQL Server Web 站点 (www.microsoft.com)

· Oracle 数据库: <u>Oracle 数据库 JDBC</u> (http://www.oracle.com/technetwork/database/features/jdbc/ index.html)

**注:** 您要记住的一个重要事项是, JDBC 数据提供程序可以远程监视数据库服务器。Java 运行时环境以及所 连接数据库服务器的 JDBC 驱动程序 JAR 文件必须位于运行代理程序的系统上。

以下 Java 版本受支持:

·Oracle Corporation Java V5 或更高版本

· IBM Corporation Java V5 或更高版本

### 过程

1. 在"代理程序初始数据源"页面或"数据源位置"页面中,单击来自服务器的数据区域中的日志数据。

- 2. 在数据源区域中,单击 JDBC。
- 3. 单击下一步。

4. 在"JDBC 信息"页面的 JDBC 信息区域中,单击浏览以连接至数据库并构建 SQL 查询。

使用 JDBC 浏览器来连接到数据库并查看其表,以使您能够构建用于收集所需数据的 SQL 查询。当您选择表和列时,系统将生成一个查询,并针对该查询所返回的每个列添加特性。您可以修改和测试所生成的查询,以确保返回所需的数据。

**注:** 您也可以手动创建 JDBC 数据源,而无需单击**浏览**。如果您想手动创建数据源,请指定查询并单击 下一步。您必须针对该查询所返回的每个列,按返回列的顺序定义特性。

您可以使用 JDBC 数据提供程序来对数据库运行 SQL 查询和存储过程以收集监视数据。在指定 SQL 查询以收集数据时,可以在 SQL 语句中包括 WHERE 子句以便对返回的数据进行过滤。SQL 语句还可以连接多个表中的数据。除了 SQL select 语句之外,JDBC 数据提供程序还可以运行存储过程。有关运行存储过程的信息,请参阅<u>第 113 页的『存储过程』</u>。

5. Browser 第一次打开时, Java Database Connectivity (JDBC) Browser 窗口将指示未选择任何连接。您必须添加连接。单击**添加**,并执行<u>添加连接的步骤</u>。

如果您已定义连接,那么将使用该连接,并且您可以前进到步骤 <u>第 110 页的『6』</u>。

注:状态字段显示当前连接的状态。

请使用以下步骤添加连接:

a) 在"JDBC 连接"页面上,单击 JDBC 连接,然后单击下一步。

b) 在"连接属性"页面上填写字段,如下所示:

#### 连接名称

JDBC 连接的名称。输入此连接的唯一名称。该名称用于引用浏览器中的连接。

**数据库类型** 数据库的类型。选择要连接到的数据库产品。例如,要连接至 IBM Db2 数据库,请选择 DB2。

用户名

必须定义为至少对数据库具有读访问权,但不必是数据库管理员

密码

必须定义为至少对数据库具有读访问权,但不必是数据库管理员

主机名

运行数据库服务器的主机的名称。使用 JDBC 可以监视远程数据库,因此您并不限于仅监视本地系统上的数据库。

#### 端口

该主机名的主机上数据库服务器正在侦听的端口。

数据库

要连接到的数据库的名称。

Jar 目录

包含用于连接到数据库的 JDBC JAR 文件的目录。请输入路径名,或者单击浏览以查找目录。

- c) 可选:如果要保存此连接的密码,请选中将密码保存在 Agent Builder 工作空间中复选框。
- d) 可选:如果您希望从这些属性复制此应用程序服务器类型的缺省值,请选中**设置为代理程序配置缺** 省值复选框。

如果您在与受监视系统类似的系统上构建代理程序,最好选中此框。如果未选中此框,配置该代理程序的用户将看到空的字段。然后用户必须确定没有缺省值的所有信息的值。

e) 单击测试连接,以创建一个使用所指定配置参数的数据库连接。 在"连接属性"页面上将显示一条消息、指示连接是否成功。

f) 当您有了正常工作的连接之后, 请单击完成。

6. 在在"Java 数据库连接 (JDBC) 浏览器"窗口中,将与配置的数据库建立连接。该数据库中包含的表将 显示在数据库表区域中。请选择数据库表,以便在选定表中的列区域中查看该表中包含的列。

注:

a. 单击望远镜图标,以便在数据库表列表中搜索表。

b. 缺省情况下将显示所有表。可以通过选择另一个过滤选项对显示的表进行过滤。<u>第 110 页的表 11</u> 显示了可用的过滤选项。

表 11. 过滤选项				
过滤选项	描述			
全部	显示所有表			
用户	只显示用户表			
系统	只显示系统表			
查看	只显示数据库视图			

**注:** 如果要检索特定列,请仅选择这些列。如果您选择表,那么 Agent Builder 将自动构建一个查询来收 集该表中所有的列,并针对当前存在于该表中的所有列创建特性。

可以通过以下方法来选择列:

- ·选择表将获得对所有列的缺省查询。
- ·选择列将只获取那些列。
- 7. 可选: 修改"特性信息"页面中为错误、缺失数据和 No 值设置的枚举值。 修改这些值以避免与可能从数据库表列返回的合法值发生任何重叠。
- 8. 可选: 在"Java 数据库连接 (JDBC) 浏览器"窗口中单击测试,以测试并修改 SQL 语句。
  - "运行 SQL 语句"窗口随即打开。
  - a) 在 SQL 语句字段中输入或修改 SQL 语句。
  - b) 单击运行以运行该 SQL 语句。
    结果将显示在"结果"区域中。请继续修改并测试该语句,直到您对返回的数据满意为止。
    c) 单击确定以保存该语句、创建正确的特性并返回到"JDBC 信息"窗口。
- 9. 可选: 在"JDBC 信息"窗口中单击测试,以便在更现实的代理程序环境中测试该属性组。有关测试 JDBC 属性组的更多信息,请参阅<u>第 114 页的『测试 JDBC 属性组』</u>。如果在此测试期间更改了 JDBC 语句,那么还必须调整特性,以确保对于该 JDBC 语句返回的每个列,都存在一个相应的特性且顺序正确。
- **10**. 可选: 通过单击**高级**,您可以创建过滤器,以限制此属性组所返回的数据。有关对来自属性组的数据进行过滤的更多信息,请参阅<u>第38页的『过滤属性组』</u>
- 11. 在"JDBC 信息"页面的操作系统部分中,选择操作系统,并单击下一步。有关应选择哪个操作系统的信息,请参阅<u>第 48 页的『指定操作系统』</u>。

**注:** 单击**插入配置属性**以选择要插入的属性。有关更多信息,请参阅<u>第 175 页的『第 10 章 定制代理程</u> 序配置』。

- 12. 在"选择键特性"页面上,选择键特性或指示此数据源仅生成一个数据行。有关更多信息,请参阅<u>第</u> 15页的『选择键属性』。
- 13. 如果您想测试先前定义的数据源,请在 Agent Editor 窗口中选择**数据源**选项卡,并选择 JDBC 数据源。 在"JDBC 属性组信息"区域中,单击测试。有关测试的更多信息,请参阅<u>第 114 页的『测试 JDBC 属</u>性组』。
- 14. 如果您想查看自动生成的配置节,请单击 Agent Editor 的**插入配置属性**选项卡。 您可以更改这些属性的标签或缺省值,以便与用户最初配置代理程序时看到的缺省值相匹配。
- 15. 可选: 填写 "特性信息"页面;有关详细信息,请参阅<u>第 34 页的『用于定义特性的字段和选项』</u>。如 果您已选择手动创建 JDBC 数据源,而不在步骤 <u>第 109 页的『4』</u>中单击 Browse,请执行此步骤。

Agent Builder JDBC 数据源支持从大多数 SQL 类型收集数据。<u>第 111 页的表 12</u> 中的信息描述了当 JDBC Browser 检测到其中一种类型的列时所创建的特性类型。这些数据类型是支持用于监视代理程序 的类型。

表 12. 支持用于监视代理程序的 SQL 数据类型				
SQL 数据类型	创建的 IBM Tivoli Monitoring 特性			
BIGINT	此数据类型是 IBM Tivoli Monitoring 中的 64 位标 尺值。如果您选择了 IBM Tivoli Monitoring V6.2 兼容性,那么这是 32 位标尺。			
DECIMALDOUBLEFLOATNUMERICREAL	这些 SQL 类型在 IBM Tivoli Monitoring 中将创建为 64 位标尺特性。如果数据库元数据包含小数位数值,那么将使用该值;否则,小数位数设为 1。如果您选择了 IBM Tivoli Monitoring V6.2 兼容性,那么此特性为 32 位标尺。			
BITINTEGERSMALLINTTINYINT	下列 SQL 类型在 IBM Tivoli Monitoring 中将创建为 32 位标尺特性。			

表 12. 支持用于监视代理程序的 SQL 数据类型 (续)			
SQL 数据类型	创建的 IBM Tivoli Monitoring 特性		
BOOLEAN	此值在 IBM Tivoli Monitoring 中是 32 位标尺,并 具有 TRUE 和 FALSE 枚举。		
TIMESTAMP	此类型的列中的数据将转换为 16 字节 IBM Tivoli Monitoring 时间戳记特性。		
TIMEDATECHARLONGVARCHARVARCHAR	浏览器将这些 SQL 类型均视为字符串特性。列大 小将用作特性大小,最大大小为 256,这是 JDBC 浏览器的缺省字符串特性大小。		

**注:** 如果从未列出的数据类型收集数据,那么缺省情况下使用字符串特性。代理程序还尝试从数据库以字符串形式收集数据。

修改"特性信息"页面中为错误、缺失数据和 No 值设置的枚举值(如果必要)。修改这些值以避免与可能从数据库表列返回的合法值发生任何重叠。

## JDBC 配置

在代理程序中定义 JDBC 数据源时,将为您创建部分配置属性。

如果在代理程序中定义了 JDBC 数据源,那么该代理程序必须使用 Java 来连接 JDBC 数据库服务器。Java 配置属性将自动添加到该代理程序中。下列 Java 配置属性特定于代理程序运行时配置:

·Java 主目录:指向 Java 安装目录的标准路径

- ·JVM 自变量:使用此参数为 Java 虚拟机指定一个可选自变量列表。
- ·跟踪级别::此参数定义要写入 Java 跟踪日志文件的信息量。缺省设置为只将错误数据写入日志文件。

**注:** Agent Builder 不需要这些 Java 属性,因为它使用自己的 JVM 以及日志记录,这是通过 JLog 插件配置 的。

如果在代理程序中定义 JDBC 数据源, 那么将向该代理程序中自动添加以下必需的公共配置字段:

- · JDBC database type: 要连接的数据库的类型,即 IBM Db2、Microsoft SQL Server 或 Oracle Database Server。
- ·JDBC user name:用于向数据库服务器认证的用户名。
- ·JDBC password:用于向数据库服务器认证的密码。
- ·Base paths:要在其中搜索 Class Path 字段所指定 JAP 文件的目录列表,或者为 JAR directories 字段所指 定的非标准目录的列表。目录名称在 Windows 系统上以分号(;)分隔,在 UNIX 系统上以分号(;)或冒 号(:)分隔。
- ·Class path:代理程序要搜索的显式指定 JAR 文件。在找到 JAR 文件之前,所有具有非标准名称的文件都 将追加到每个"基本路径"中。
- ·JAR directories: 要在其中搜索 JAR 文件的目录的列表。目录名称在 Windows 系统上以分号(;)分隔, 在 UNIX 系统上以分号(;)或冒号(:)分隔。您不必显式指示这些目录中的 JAR 文件;这些 JAR 文件会 因为它们在其中一个目录中而被找到。但不会搜索这些目录的子目录。在找到目录之前,所有具有非标准 名称的目录都将追加到每个"基本路径"中。

运行时配置还需要您指定一些其他详细信息才能连接到数据库。可以选择如何指定其余配置项,指定为 JDBC URL 或基本配置属性(缺省设置):

·URL 配置选项

- JDBC 连接 URL: 特定于供应商的连接 URL, 提供有关数据库所在的主机以及连接到的端口号的详细信息。URL 格式通常如下所示:

jdbc:identifier://server:port/database

请参阅 JDBC 驱动程序供应商文档以了解不同的 URL 格式。

·JDBC 基本属性选项(缺省设置)

JDBC 服务器名称:正在运行数据库服务器的主机名。 JDBC 数据库名称:建立了连接的主机上的数据库的名称。 JDBC 端口号:数据库服务器正在侦听的端口号。

**注:** 借助 JDBC 数据提供程序,可以使用子节点在同一个代理程序中监视多种数据库类型。要使用该方法进行监视,必须仔细定义"子节点配置覆盖"。如果监视多个数据库类型,以下配置设置很可能不同:

- ·JDBC 数据库类型
- ·JDBC 用户名
- ·JDBC 密码

如果您正在使用基本配置选项,那么还必须在"子节点配置覆盖"页面上针对下列属性定义覆盖项:

- ·JDBC 服务器名称
- ·JDBC 端口号
- ·JDBC 数据库名称

要为子节点定义配置覆盖项,请参阅<u>第159页的『第9章使用子节点』</u>,以获取有关访问"**子节点配置覆** 盖"页面的更多详细信息。当您在运行时配置代理程序时,必须针对创建的每个新子节点实例配置所有这些 属性。

除了配置覆盖项,您的代理程序还必须指向各个数据库类型的 JDBC 驱动程序,您将在子节点中连接到该驱动程序。*JAR* 目录参数是指向 JDBC 驱动程序最方便的方式。使用分号来分隔每个目录,以列出包含 JDBC 驱动程序的目录。例如,如果要使用代理程序连接到 Db2 和 Oracle 数据库,您必须指定类似于以下示例的 *JAR* 目录值: C:\Program Files\IBM\SQLLIB\java;C:\oracle\jdbc。

# 存储过程

可以与 JDBC 数据提供程序配置使用的示例 SQL 和 Db2 存储过程。

JDBC 数据提供程序可以处理存储过程返回的结果集。字符串或整数输入参数可以传递给存储过程。以下语 法运行存储过程:

call[:index] procedureName [argument] ...

#### 其中:

#### index

可选整数,指定数据提供程序将使用哪个结果集。当存储过程返回多个结果集而您只需要收集其中一个结果集中的值时,此参数很有用。如果不指定 index,将收集并返回每个结果集的数据。

#### procedureName

JDBC 数据提供程序将运行的存储过程的名称。

#### argument

存储过程的输入参数。必须使用空格分隔多个自变量。如果自变量包含空格字符,请使用双引号将整个 自变量括起来。如果自变量可以解析为整数,那么它将作为整数自变量传递到存储过程。任何括在双引 号内的自变量都将作为字符串自变量进行传递。

#### SQL Server 样本 call sp\_helpdb

运行过程 call sp\_helpdb,此过程不需要自变量。返回的所有结果集中的数据都将包括在数据提供程序 所返回的数据中。

### call:2 sp\_helpdb master

使用 master 自变量运行过程 sp\_helpdb。此自变量是一个字符串输入自变量。只有此存储过程所返回的第 二个结果集中的数据,才会包括在数据提供程序所返回的数据中。

如果不指定索引,将收集返回的所有结果集中的数据。必须确保这些情况下返回的数据与您定义的属性兼容。Agent Builder 根据返回的第一个结果集创建属性,任何其他结果集都应该与第一个结果集兼容。

#### Db2 存储过程

以下是以 SQL 编写的样本 Db2 函数。此函数演示如何返回可以由 Agent Builder JDBC 数据提供程序处理的 结果:

```
-- Run this script as follows:
-- db2 -td# -vf db2sample.sql
-- Procedure to demonstrate how to return a query from
-- a DB2 stored procedure, which can then be used by
-- an Agent Builder JDBC provider. The stored procedure
-- returns the following columns:
-- Name
                         Description
                                                       Data Type
-- current_timestamp The current system time timestamp

-- lock_timeout The lock timeout numeric s

-- user The user for the session String 12
                                                    numeric scale 0
-- user
                         The user for the session String 128 characters long
DROP procedure db2sample#
CREATE PROCEDURE db2sample()
  RESULT SETS 1
  LANGUAGE SQL
BEGIN ATOMIC
  -- Define the SQL for the query
DECLARE c1 CURSOR WITH HOLD WITH RETURN FOR
  SELECT CURRENT TIMESTAMP as current_timestamp
CURRENT LOCK TIMEOUT as lock_timeout, CURRENT USER as user
  FROM sysibm.sysdummy1;
  -- Issue the query and return the data
  OPEN c1;
END#
```

从 Agent Builder 中,可以使用与针对其他存储过程定义的相同语法来调用此函数。在这种情况下,请将 call db2sample 定义为 JDBC 语句以运行此存储过程。

#### Oracle 存储过程

Oracle 存储过程不返回结果集。您必须编写返回 Oracle 引用游标的函数。以下是以 PL/SQL 编写的样本 Oracle 函数,此函数演示如何返回可以由 Agent Builder JDBC 数据提供程序处理的结果:

```
CREATE OR REPLACE FUNCTION ITMTEST
RETURN SYS_REFCURSOR
IS v_rc SYS_REFCURSOR;
BEGIN OPEN v_rc FOR SELECT * FROM ALL_CLUSTERS;
RETURN v_rc;
END;
```

从 Agent Builder 中,可以使用与针对其他存储过程定义的相同语法来调用此函数。在这种情况下,请将 call ITMTEST 定义为 JDBC 语句来运行此存储过程。由于该 Oracle 函数必须返回光标引用,所以 Oracle 函数只能处理一个结果集。这意味着 Oracle 不支持索引选项,因为无法返回多个结果集。

## 测试 JDBC 属性组

您可以对在 Agent Builder 中创建的 JDBC 属性组进行测试。

#### 过程

1. 可以通过以下方法启动测试过程:

- · 在创建代理程序期间,单击"JDBC 信息"页面上的测试。
- · 在创建代理程序之后,在 Agent Editor 的"数据源定义"页面上选择属性组,然后单击测试。有关 Agent Editor 的更多信息,请参阅第 17 页的『第 4 章 使用 Agent Editor 修改代理程序』。

在上述两个步骤中的某个步骤中单击测试之后,将显示"测试 JDBC 语句"窗口。

2. 可选: 在开始测试之前, 您可以设置环境变量、配置属性和 Java 信息。

有关更多信息,请参阅<u>第191页的『属性组测试』</u>。有关 JDBC 配置属性的更多信息,请参阅(<u>第112</u>页的『JDBC 配置』)。

3. 单击启动代理程序。

此时将显示一个窗口,表明代理程序正在启动。

4. 要模拟来自 Tivoli Enterprise Portal 或 SOAP 的代理程序数据请求,请单击**收集数据**。

代理程序会使用指定的 SQL 查询对数据库进行查询。"测试 JDBC 语句"窗口会收集并显示自代理程序 上次启动以来,代理程序高速缓存中的全部数据。

**注:** 返回数据的顺序至关重要;例如,第一个返回列中的数据值始终分配给第一个属性。如果更改 JDBC 语句,那么必须对属性进行添加、移除或重新排序,以与此语句返回的列相匹配。

5. 可选: 如果返回的数据未如预期,请单击检查结果。

"**数据收集状态**"窗口将会打开,并显示有关该数据的更多信息。<u>第 239 页的『Performance Object</u> Status 节点』中描述了 Data Collection Status 窗口收集和显示的数据

6. 通过单击停止代理程序停止代理程序。

7. 单击确定或取消以退出"测试 JDBC 语句"窗口。单击确定将保存您所作的全部更改。

#### 相关概念

<u>第191页的『在Agent Builder 中测试代理程序』</u> 使用 Agent Builder 创建代理程序之后,可以在 Agent Builder 中测试该代理程序。

# 使用 Ping 来监视系统可用性

您可以定义数据源以使用因特网控制报文协议 (ICMP) 回传 Ping 测试网络设备列表。要测试的设备的主机名 或 IP 地址列示在一个或多个设备列表文件中。单独的 Ping 配置文件指定每个设备列表文件的路径。然后, 在代理程序运行时配置中设置 Ping 配置文件的名称。这些结果包括每个网络设备的状态。

#### 开始之前

创建设备列表文件和 Ping 配置文件(请参阅<u>第 116 页的『配置文件』</u>)。

### 关于此任务

网络管理其中一部分职能便是能够决定系统是否对因特网控制报文协议 (ICMP) Ping 进行回应。使用此数据 源以监视环境中一组服务器或其他关键设备的基本联机或脱机状态。用 Ping 进行监视不仅简单,而且开销 低。要监视一列设备,请向代理程序添加 Ping 数据收集器。

### 过程

- 1. 在"代理程序初始数据源"页面或"数据源位置"页面中,单击监视数据类别区域中的网络管理数据。
- 2. 在数据源区域中, 单击 Ping。
- 3. 单击下一步。
- 4. 在 Ping 信息窗口中的操作系统区域中,选择操作系统。
- 5. 可选: 您可以通过单击**测试**来测试此属性组。有关测试的更多信息,请参阅<u>第 117 页的『测试 Ping 属性</u> 组』
- 6. 可选:通过单击高级,您可以创建过滤器,以限制此属性组所返回的数据。有关过滤来自属性组的数据 的更多信息,请参阅<u>第 38 页的『过滤属性组』</u>
- 7. 执行下列其中一个步骤:
  - a) 如果您正在使用"代理程序"向导,请单击下一步。
  - b) 单击完成,以保存该数据源并打开 Agent Editor。
- 8. 有关添加属性的更多信息, 请参阅(<u>第32页的『创建属性』</u>)。

### 结果

有关 Ping 属性组的更多信息,请参阅 第 268 页的『Ping 属性组』。

# 配置文件

使用配置文件为代理程序提供要执行 Ping 操作的设备的列表。

代理程序需要两种类型的配置文件。

#### 设备列表文件

包含要执行 Ping 操作的设备的列表。如果有许多设备,您可以在多个设备列表文件中对其进行划分。代 理程序对每个设备列表文件启动单独的线性并且以并行方式循环通过这些文件。该线程以 60 秒的时间间 隔循环通过每个文件,或以 30 秒加上 Ping 设备列表所花时间的时间间隔循环通过,以两者中时间间隔 长的为准。

设备列表文件的语法如下所示:

LISTNAME=list\_name device\_name or host\_name device\_name or host\_name device\_name or host\_name device\_name or host\_name

其中, list\_name 是该文件中的设备的描述。如果未定义任何列表名称, 那么将使用设备列表文件的名称。列表名称不需要是文件中的第一个条目。但是, 如果文件有多个列表名称定义, 那么使用最后的定义。

设备列表文件中可以包含的设备数量没有限制。然而,包含过多条目则无法实现定义关键设备目标列表的目的,并会增加总的工作负载。在 60 秒监视时间间隔内检索每个设备的状态可能更难。

在每个循环开始时,代理程序会检查设备列表文件的最后修改时间。如果该文件的最近一次修改时间晚 于代理程序最近一次读取该文件的时间,那么代理程序将重新读取该文件,而不需要重新启动。

#### Ping 配置文件

指定每个设备列表文件的位置。使用 Ping 配置文件位置的标准路径或相对路径。Ping 配置文件作为运行时配置参数传递到代理程序。

#### 示例

在以下示例中,设备分为两个文件。 /data/retailList.txt文件包含以下条目:

LISTNAME=Retail frontend.mycompany.com productdb.mycompany.com

/data/manufacturingList.txt 文件包含以下条目:

```
LISTNAME=Manufacturing systems
manufloor.mycompany.com
stats.supplier.com
```

Ping 文件 /data/pinglists.txt 包含以下条目:

/data/retailList.txt /data/manufacturingList.txt

#### "网络管理"配置属性

添加 Ping 数据源之后, 配置将显示在 Agent Editor 的"运行时配置信息"页面上。

"运行时配置信息"页面的网络管理配置部分包含以下属性:

表 13. "网络管理"配置属性					
姓名	有效值	是否必需	描述		
Ping 配置文件	文件的路径	否。如果未提供此文件, 那么将从代理程序 bin 目 录使用 KUMSLIST 文 件。	包含一列文件(每个文件 都包含一列要使用 ICMP Ping 监视的主机)的文 件的路径。		

# 测试 Ping 属性组

您可以对在 Agent Builder 中创建的 Ping 属性组进行测试。

# 过程

- 1. 可以通过以下方法启动测试过程:
  - · 在创建代理程序期间, 单击 "Ping 信息"页面上的测试。
  - · 在创建代理程序之后,在 Agent Editor 的"数据源定义"页面上选择属性组,然后单击测试。有关 Agent Editor 的更多信息,请参阅第 17 页的『第 4 章 使用 Agent Editor 修改代理程序』。

在上述两个步骤中的某个步骤中单击测试之后,将打开"测试设置"窗口

- 2. 可选: 在开始测试之前, 您可以设置环境变量和配置属性。有关更多信息, 请参阅<u>第191页的『属性组</u> 测试』。
- 3. 单击浏览以选择 Ping 配置文件。有关 Ping 配置文件的更多信息,请参阅 第 116 页的『配置文件』
- 4. 单击启动代理程序。此时将显示一个窗口,表明代理程序正在启动。
- 5. 要模拟来自监视环境的代理程序数据请求,请单击**收集数据**。代理程序会对 Ping 配置文件中引用的设备 列表文件中所指定的设备执行 Ping 操作。
- 6. "测试设置"窗口会收集并显示自代理程序上次启动以来,代理程序高速缓存中的全部数据。
- 7. 可选:如果返回的数据未如预期,请单击检查结果。 "数据收集状态"窗口将会打开,并显示有关该数据的更多信息。<u>第 239 页的『Performance Object</u>
  - <u>Status 节点</u>。中描述了 Data Collection Status 窗口收集和显示的数据。
- 8. 通过单击停止代理程序停止代理程序。
- 9. 单击确定或取消以退出"测试设置"窗口。单击确定将保存您所作的全部更改。

# 相关概念

<u>第191页的『在Agent Builder 中测试代理程序』</u> 使用 Agent Builder 创建代理程序之后,可以在 Agent Builder 中测试该代理程序。

# 监视 HTTP 可用性和响应时间

您可以配置数据源以监视所选 URL 的可用性和响应时间。使用配置文件来定义 URL 列表。在代理程序运行时配置中设置文件名称。在 IBM Tivoli Monitoring 中,也可以使用"执行操作"命令来添加和移除受监视的 URL。每个 URL 的状态作为一行添加到所产生的数据集中。

# 关于此任务

对于所监视的每个 URL,结果将提供有关对 HTTP 请求作出的 HTTP 响应的一般信息。结果包括能否对其进行检索、检索所耗时间以及响应大小。如果响应内容为 HTML,那么还将提供有关 URL 中的页面对象的信息。

您可以监视使用 HTTP、HTTPS、FTP 和文件协议的 URL。请在 HTTP URL 文件中指定所要监视的 URL,或者通过"执行操作"选项进行指定。

**要点:** 在发布时, "执行操作"命令在 IBM Cloud Application Performance Management 环境中不可用。它 们仅在 Tivoli Monitoring 环境中可用。

此数据源需要 Java 运行时环境。以下 Java 版本受支持:

- ·Oracle Corporation Java V5 或更高版本
- · IBM Corporation Java V5 或更高版本

使用以下过程来创建用于监视 URL 列表的属性组:

# 过程

1. 在"代理程序初始数据源"页面或"数据源位置"页面中,单击来自服务器的数据区域中的日志数据。 2. 在数据源区域中,单击 HTTP。 3. 单击下一步。

4. 在"HTTP 信息"页面上的操作系统区域中,选择一个或多个操作系统。

5. 可选: 单击测试以测试此属性组。有关测试的更多信息,请参阅第 123 页的『测试 HTTP 属性组』。

- 6. 可选: 单击**高级**以创建过滤器,从而限制此属性组所返回的数据。有关对来自属性组的数据进行过滤的 更多信息,请参阅<u>第 38 页的『过滤属性组』</u>
- 7. 执行下列其中一个步骤:

a) 如果您正在使用"代理程序"向导,请单击下一步。

b) 单击完成, 以保存该数据源并打开 Agent Editor。

#### 结果

HTTP 数据源创建两个属性组: "受管 URL"和"URL 对象"。您可以添加、修改或删除属性。 相关任务

第32页的『创建属性』

您可以向数据集添加新属性。

#### 相关参考

<u>第 270 页的『HTTP 属性组』</u> 受管 URL 和 URL 对象这两个 HTTP 属性组用于接收来自 URL 及其中对象的信息。

## HTTP 表

有关缺省 HTTP 属性组的参考信息。

HTTP 数据源所创建的两个属性组为:

#### 受管 URL

"受管 URL"表提供有关每个受监视 URL 的可用性和响应时间数据。

#### URL 对象

对于嵌入的每个对象, "URL 对象"表都包含一个单独的 URL 条目。例如, "受管 URL"报告中所列的 Web 站点中可能使用的 .gif 和 .jpg 文件。

有关 "受管 URL" 和 "URL 对象" 表中使用的语法的信息, 请参阅 第 119 页的 『HTTP 属性的特定字段』。

如果要监视 Web 站点中特定对象的响应时间和可用性,请查看"URL 对象"表的内容。"URL 对象"表用 于监视在下载的 HTML 文件中检测到的对象的特定列表。下表列示要在其中搜索所要监视的对象的 HTML 元 素以及这些元素中引用了对象的属性:

表 14. 要在其中搜索所要监视的对象的 HTML 元素				
HTML 元素	包含所要监视的对象的属性			
img	src			
script	src			
embed	src			
object	codebase or data			
body	background			
input	src			

在以下 HTML 片段示例中, 受监视对象是 img 元素的 src 属性所引用的图像。

<img src="/v8/images/id-w3-sitemark-simple.gif" alt="" width="54" height="33" />

此图像的完整 URL 将根据源文档的 URL 计算。

注:如果您不想监视在网页中找到的对象,请在"URL监视"配置部分中,将页面对象收集属性设置为否。

# HTTP 属性的特定字段

在"**属性信息**"页面中,有两个用于 HTTP 属性的字段,它们定义从 URL 收集数据的方式。**属性类型**字段可以是列表中的任意值,用于控制所返回的 URL 相关信息。某些属性类型要求**类型值**字段包含值。

下表描述"受管 URL"属性组的所有属性类型及类型值(如果需要):

表 <i>15. HTTP</i> 属性信息 - 受管 URL							
属性类型	描述	类型值	返回的数据类型	与 FTP 和文件协议 的差别			
XPath Query	对通过 URL 连接返 回的内容运行 XPath 查询。此查 询必须编写为返回 对属性有用的数 据,而不是节点列 表。	要对通过 URL 连接 获取的内容运行的 XPath 查询。	返回的数据可以是 字符串、数字或时 间戳记值。如果数 据采用 XML 日期时 间格式,那么您可 以将时间戳记指定 为属性类型。代理 程序将值转换为 Candle Timestamp。	无			
Response Time	从所请求的 URL 下 载内容所耗用的时 间(毫秒)。	无	整数(毫秒数)	无			
Response Message	服务器返回的 HTTP 响应消息。	无	String	仅当此 URL 使用 HTTP 或 HTTPS 协 议时,响应消息才 适用。			
Response Code	服务器返回的 HTTP 响应代码。	无	整数	仅当此 URL 使用 HTTP 或 HTTPS 协 议时,响应代码才 适用。对于文件或 FTP URL,此代码 始终为 0。			
Response Length	从所请求的 URL 下 载的内容的大小 (以字节计)	无	整数(以字节计的 大小)	无			
Response Header	可用于从其中一个 URL响应头字段检 索值的响应头。自 变量指定所要请求 的字段。	要收集的响应头。	String	通常,FTP和文件 协议没有任何可收 集的头。			
Request URL	这是所连接的 URL。所有响应关 键字都将提供有关 与此 URL 的连接的 信息。您可以使用 XPath 查询从通过 访问此 URL 返回的 内容中获取信息。	无	String	无			

表 15. HTTP 属性信息 - 受管 URL (续)					
属性类型	描述	类型值	返回的数据类型	与 FTP 和文件协议 的差别	
页面对象数	在受监视 HTML 页 面上发现的受 "URL 对象"属性 组监视的对象数。	无	整数	无	
对象总大小	在此 Web 页面的 "URL 对象"属性 组中监视的对象的 大小总计。	无	整型(字节)	无	
别名	用户对此 URL 指定 的别名。	无	String	无	
用户	用户对此 URL 指定 的数据。	无	String	无	

下表描述 "URL 对象" 属性组的属性类型:

表 16. HTTP 属性信息 - URL 对象					
属性类型	描述	类型值	返回的数据类型	与 FTP 和文件协议 的差别	
URL	在"受管 URL"表 中监视的 URL。	无	String	无	
对象名称	在 HTML 页面中监 视的对象的 URL。	无	String	无	
对象大小	从对象名 URL 下载 的内容的大小(以 字节计)。	无	数字	无	
对象响应时间	下载页面对象所用 的时间(毫秒)。	无	数字	无	

# 监视 URL

通过将 URL 包括在 URL 文件中,或者通过使用"HTTP URL 添加执行操作"选项,可以对任何 URL 启动监视。

# URL 文件

配置中指定的 URL 文件可以位于任意目录中。如果此文件不存在或为空,那么您可以使用"执行操作"来启动 URL 监视。有关更多信息,请参阅<u>第 121 页的『"执行操作"选项』</u>。如果您已有使用 Tivoli Universal Agent HTTP Data Provider 的 Tivoli Universal Agent,那么可以复用 KUMPURLS 文件。配置此代理程序时,请指向 KUMPURLS 文件。

下表根据添加 URL 的方法提供如何将 URL 输入到 URL 文件的示例。

表 17. URL 文件条目					
URL	添加者				
www.bbc.co.uk http://weather.com www.ibm.com	以手动方式在此文件中添加条目。如果未指定协议 (如 www.ibm.com 示例所示),那么将采用 HTTP。				

表 17. URL 文件条目 (续)				
URL	添加者			
<pre>ftp://userid:password@ftpserver/ index.html</pre>	使用文件传输协议 (FTP) 以手动方式添加			
http://www.ibm.com USER=ibm ALIAS=ibm	使用"HTTP URL 添加"执行操作			
file:/tmp/samples.html USER=samples \ ALIAS=samples	使用通过 FTP 执行的"HTTP URL 添加"执行操作			
http://google.com INTERVAL=60 CACHE=50 \ USER=google ALIAS=search	Tivoli Universal Agent KUMPURLS 文件中的示例			

如果您直接编辑 URL 文件,那么所作的更改将在此代理程序下次收集数据时实现。

# "执行操作"选项

另外,还可以通过名为"HTTP URL 添加"的执行操作选项指定要监视的 URL。

**限制:** 此选项在 IBM Cloud Application Performance Management 的当前发行版中不可用,这是因为,您无 法手动启动"执行操作"命令。

选择此选项后,系统将显示一个窗口,您可以在该窗口中指定下列参数:

#### URL

这是一个必需参数,用于表示 URL 自身。您可以在指定或不指定 http:// 或 https:// 前缀的情况下 输入此参数。

### 别名

这是一个可选参数,您可以通过指定此参数使更有意义的名称与 URL 相关联。在此参数中,不允许输入 空格。如果未填写此参数,那么"别名"将缺省为空。

#### User\_Data

这是一个可选参数,您可以通过指定此参数输入关于 URL 的数据。如果未填写此参数,那么 User\_Data 将缺省为 INITCNFG。

填写信息并关闭窗口后,请对与代理程序相关联的目标受管系统指定 HTTP URL 添加操作。针对新 URL 进行的监视将立即开始。此 URL 还将添加到 URL 文件中,因此,代理程序重新启动后将继续对其进行监视。

相应的"执行操作"选项名为 HTTP URL 移除。使用 HTTP URL 移除操作立即停止对特定 URL 的监视。 这还会从 URL 文件中删除所移除的 URL。"**HTTP URL 移除**"窗口仅要求提供 URL 和 User\_Data 值。URL 和 User\_Data 值必须与 Tivoli Enterprise Portal 中显示的值匹配,否则"移除"操作将失败。例如,即使 在"添加"操作的 URL 字段中省略了 http://,也必须将其包括在"移除"操作的 URL 字段中。如果未指 定 User\_Data,那么必须按 Tivoli Enterprise Portal 中显示的那样指定 INITCNFG。

如果已将 URL 手动添加到 URL 文件中,那么可使用"执行操作"将其删除。如果使用"执行操作"进行删除,那么必须按 Tivoli Enterprise Portal 中显示的那样指定值。例如,如果在 URL 文件中添加了www.ibm.com,那么 Tivoli Enterprise Portal 将显示 http://www.ibm.com 作为 URL,并显示 INITCNFG 作为 User\_Data。要使用"执行操作"来移除此 URL,必须使用 Tivoli Enterprise Portal 中显示的值。

填写信息并关闭窗口后,请对与代理程序相关联的目标受管系统指定 HTTP URL 移除操作。

# 监视 https:// URL

HTTP 数据源仅监视不需要脚本式访问或交互式提示的安全的 https:// URL。

如果可使用标准 HTTP Get 调用来检索 https:// URL, 那么可对其进行监视。

# 代理服务器

如果正在运行代理程序的系统需要代理才能访问 SOAP 数据提供程序,那么必须指定代理服务器配置属性。 有关更多信息,请参阅<u>第 122 页的『代理服务器配置』</u>。

# HTTP 配置

有关 HTTP 配置的参考信息。

添加 HTTP 数据源之后,配置将显示在 Agent Editor 的"运行时配置"页面上。对于 URL 监视、代理服务器 认证和 Java,添加了相应的配置部分。

# "URL 监视"配置

"URL 监视"配置节包含以下属性:

表 18. "URL 监视"配置属性						
姓名	有效值	是否必需	描述			
HTTP URL 文件	文件的路径	是	包含 URL 列表的文件的 路径。			
页面对象收集	"是"和"否" 缺省值为"是"。	否	是否下载在 Web 页面中 找到的对象并从中收集数 据。			

# 代理服务器配置

代理服务器配置节包含以下属性:

表 19. 代理服务器配置属性					
姓名	有效值	是否必需	描述		
代理主机名	String	否	要用于 HTTP 连接的代理 主机名。		
代理用户名	String	否	代理服务器的用户名。		
代理端口	正整数 缺省值为 80。	否	代理服务器的 HTTP 端口 号。		
代理密码	密码	否	代理服务器的密码。		

注:如果代理主机名属性为空白,那么将不使用代理。

# Java 配置

如果在代理程序中定义了 HTTP 数据源,那么该代理程序必须使用 Java 来连接到 HTTP 服务器。Java 配置 属性将自动添加到代理程序中。下列 Java 配置属性特定于代理程序运行时配置。Agent Builder 不需要这些 Java 属性,因为它使用自己的 JVM 以及日志记录,这是通过 JLog 插件配置的:

表 20. Java 配置属性						
姓名	有效值	是否必需	描述			
Java 主目录	目录的标准路径	否	指向 Java 安装目录的标 准路径。			
跟踪级别	选项 (缺省值为"错误")	是	请使用此属性来指定 Java 提供程序所使用的 跟踪级别。			

表 20. Java 配置属性 (续)					
姓名	有效值	是否必需	描述		
JVM 自变量	String	否	请使用此属性来指定 Java 虚拟机的可选自变 量列表。		

# 测试 HTTP 属性组

您可以对在 Agent Builder 中创建的 HTTP 属性组进行测试。

# 过程

1. 可以通过以下方法启动测试过程:

- · 在创建代理程序期间, 单击"HTTP 信息"页面上的测试。
- · 在创建代理程序之后,在 Agent Editor 的"数据源定义"页面上选择属性组,然后单击测试。有关 Agent Editor 的更多信息,请参阅第 17 页的『第 4 章 使用 Agent Editor 修改代理程序』

在上述两个步骤中的某个步骤中单击测试之后,将显示"HTTP测试"窗口。

- 2. 单击浏览以选择 HTTP URL 文件。有关 URL 文件的更多信息,请参阅<u>第 120 页的『URL 文件』</u>。
- 3. 可选: 在开始测试之前,请设置环境变量、配置属性及 Java 信息。 有关更多信息,请参阅<u>第 191 页的『属性组测试』</u>。有关 HTTP 配置的更多信息,请参阅 <u>第 122 页的</u> <u>『HTTP 配置』</u>。
- 4. 单击启动代理程序。

此时将显示一个窗口,表明代理程序正在启动。

- 5. 要模拟来自 Tivoli Enterprise Portal 或 SOAP 的代理程序数据请求,请单击**收集数据**。 代理程序会监视 HTTP URL 文件中定义的 URL。"**HTTP 测试**"窗口将显示所返回的全部数据。
- 6. 可选:如果返回的数据未如预期,请单击检查结果。
   "数据收集状态"窗口将会打开,并显示有关该数据的更多信息。<u>第 239 页的『Performance Object Status 节点』</u>对 Data collection Status 窗口所收集并显示的数据作了描述。
- 7. 通过单击停止代理程序停止代理程序。
- 8. 单击确定或取消以退出"HTTP测试"窗口。单击确定将保存您所作的全部更改。

### 相关概念

<u>第 191 页的『在 Agent Builder 中测试代理程序』</u> 使用 Agent Builder 创建代理程序之后,可以在 Agent Builder 中测试该代理程序。

# 监视来自 SOAP 或其他 HTTP 数据源的数据

您可以定义数据源,以接收来自 HTTP 服务器的数据(例如,使用 SOAP 协议来接收数据)。该数据源将 HTTP 请求发送到 URL,并将 XML、HTML 或 JSON 格式的响应解析为所产生数据集的属性。您可以选择要 从该请求中检索的数据。

# 关于此任务

通过使用 SOAP 数据源,您可以指定 HTTP URL 并发送 GET、POST 或 PUT 请求。对于 POST 或 PUT 请求,您可以指定关联的 POST 数据。系统将检索并解析 XML、HTML 或 JSON 响应,并通过属性将数据呈现给监视环境。您可以将属性定义为特定元素中的所有值。您还可以定义定制 XPath 值,以指定如何填充各个属性。也可以结合使用这两种机制。

使用以下过程从 URL 中收集并解析 XML、HTML 或 JSON 响应:

# 过程

1. 在"代理程序初始数据源"页面或"数据源位置"页面中,单击来自服务器的数据区域中的日志数据。

- 2. 在数据源区域中,单击 SOAP。
- 3. 单击下一步。
- 4. 在 "**SOAP** 信息"页面上,输入 URL。 缺省值为:

http://\${KQZ\_HTTP\_SERVER\_NAME}:\${KQZ\_HTTP\_PORT\_NUMBER}

**注:** 可以使用一个或多个解析为 URL 的配置变量。单击插入配置属性以选择要插入的属性。有关更多信息,请参阅第 175 页的『第 10 章 定制代理程序配置』。

5. 选择请求类型。缺省请求类型为 Get。对于 Post 和 Put 请求,请输入要处理的数据。

**注:** 对于 Post 和 Put 请求,将启用**插入配置属性**。单击**插入配置属性**以在要处理的数据中包括配置变量。有关更多信息,请参阅<u>第 175 页的『第 10 章 定制代理程序配置』</u>。

6. 单击**浏览**。

注: 如果输入 URL 并选择了请求类型,但不想使用 SOAP 浏览器来构建定义,请输入行选择 XPath。您可在"SOAP 信息"窗口中输入行选择 XPath。然后为属性组定义所有属性。

- 7. 在"SOAP 浏览器"窗口中,执行下列步骤:
  - a) 输入 URL 和选择请求类型(如果尚未执行此操作)。
  - b) 单击配置以设置在 URL 或其他字段中引用的任何配置属性。
  - c) 单击连接, 以从 SOAP 提供者获取数据。

连接至 URL 时, 文档对象模型 (DOM) 树将显示此 URL 的 XML 元素列表。HTML 或 JSON 响应将转换为 XML, 并显示为 DOM 树。有关将 JSON 响应转换为 XML 的详细信息,请参阅<u>第 127 页的</u> <u>『JSON 数据的 XML 表示』。在(第 125 页的图 14</u>)中的 WebSphere Application Server 示例中,输入了以下 URL:

http://nc053011.tivlab.raleigh.ibm.com:9080/wasPerfTool/servlet
/perfservlet?module= \threadPoolModule

这将显示 XML 元素 PerformanceMonitor。此元素是 SOAP 提供者返回的 XML 文档中的顶级 XML 元素。

😨 SOAP Browser						X
SOAP Browser Enter a URL that will return xml formatted data						
URL rfTool/servlet/perfservlet?module=	ethread Connect Insert Configuration Prope	erty	XML Attributes	Value		
Row Selection XPath		] (			Insert Configuratio	n Property
Name	Attribute Type	Type V	/alue			
						Add
Ō					Conf	iguration Cancel

# 图 14. "SOAP 浏览器"窗口

d) 在 DOM 树中,查找并选择要设置为行选择 XPath 的 XML 节点。

在<u>第 126 页的图 15</u>中的 WebSphere Application Server 示例中,选择了 PerformanceMonitor/Node/Server/Stat/Stat/Stat 节点。此节点表示属性组中的一行数据。在 DOM 树中选择节点并单击添加时,您将获得在树中该节点上定义的所有属性和元素。(单击代理程序属性区域中的添加。)

选择节点时,XML 属性区域将显示为所选节点定义的任何 XML 属性。选择 XML 属性,然后单击添加以将此属性包含在"代理程序属性"列表中。

**注:** 如果预期会有多行数据,那么 XPath 必须映射至节点集。当 Row Selection XPath 返回仅设置了 一个项的节点时,属性组仅包含一行。

😨 SOAP B	Browser					23
SOAP B Enter a U	Frowser JRL that will return xml formatted data					
URI	http://\$(KOZ HTTP SERVER N/ Conn	Insert Configuration Prope	XML Attributes			
GET 🔻		Insert Configuration Prope	Name	Value		
GET						
	٠	·				
Row Selec	tion XPath				Insert Configu	ration Property
Agent At	ttributes	Tura	Tura Malua			
Name	Attribute	туре	Type value			
						Add
						Remove
						Configuration
2					OK	Cancel
						Cancer

# 图 15. "SOAP 浏览器"窗口

e) 单击"代理程序属性"区域中的添加。

此时会显示代理程序属性列表,并填充了行选择 XPath 字段。

每个代理程序属性的 XPath 用于将 XML 节点或元素映射到代理程序属性。在<u>第 127 页的图 16</u>中的 WebSphere Application Server 示例中,代理程序属性列表中的第一个属性 Stat 无用,会将其移 除。

您可以在**类型值**字段中编辑代理程序属性的名称和 XPath。有关使用 XPath 的更多信息,请参阅 <u>第</u>129 页的『XPath 选项』

😨 SOAP Browser					
SOAP Browser Enter a URL that will return xml formatted da	ata				
			~XML Attributes		
URL rfTool/servlet/perfservlet?modu	le=thread Connect	Insert Configuration Property	Name	Value	
CET W		Tanant Canformation Descents	name	Default	
	<u>&gt;</u>	Insert Computation Property			
PerformanceMonitor					
🖮 Node					
- Server					
🖮 Stat					
😑 Stat					
BoundedRangeStatist	ic				
. Stat					
. Stat					
⊕ Stat					
BoundedRangeStatistic					
Row Selection XPath //Stat					Insert Configuration Property
IBM Tivoli Monitoring Attributes					
Name	Attribute Type	Type	Value		<b>^</b>
Stat	XPath Query				
name	XPath Query	/@nar	ne		
ID	XPath Query	/Bound	dedRangeStatistic/@ID		
highWaterMark	XPath Query	/Bound	dedRangeStatistic/@highW	/aterMark	Add
lastSampleTime	XPath Query XPath Query	/Bound	ledRangeStatistic/@integr ledRangeStatistic/@lastSa	an ann an	Remove
lowWaterMark	XPath Query	/Bound	dedRangeStatistic/@lowWa	aterMark	
lowerBound	XPath Query	/Bound	dedRangeStatistic/@loweri	Bound	
mean	XPath Query	/Bound	dedRangeStatistic/@mean		
name0	XPath Query	/Bound	dedRangeStatistic/@name		~
	VD-45 O	(D	J_JD0111111/@111117	Fire =	
0					Configuration OK Cancel

图 16. "SOAP 浏览器" 窗口

f) 在 "SOAP 浏览器" 窗口中, 单击确定以保存更改并返回到 "SOAP 信息" 窗口。

- 8. 在"SOAP 信息"窗口中,单击下一步。
- 9. 如果您之前未使用**浏览**,而是在"SOAP 信息"窗口中输入了 URL 和行选择 XPath,那么将显示"属性信息"页面。在"属性信息"页面上指定第一个属性的信息,然后单击完成。然后,可以使用 Agent Editor 指定更多属性。有关创建属性的更多信息,请参阅(<u>第 32 页的『创建属性』</u>)。
- 10. 如果您在步骤 <u>第 124 页的『6』</u>中使用了**浏览**功能,那么将显示"**选择键属性**"页面。在"**选择键属** 性"页面上,选择键属性或指示此数据源仅生成一个数据行。有关更多信息,请参阅<u>第 15 页的『选择</u> 键属性』。
- 11. 可选: 您可以通过单击**测试**来测试此属性组。有关测试的更多信息,请参阅<u>第 130 页的『测试 SOAP</u> 属性组』
- **12.** 可选: 通过单击**高级**,您可以创建过滤器,以限制此属性组所返回的数据。有关对来自属性组的数据进行过滤的更多信息,请参阅第 38 页的『过滤属性组』
- 13. 执行下列其中一个步骤:
  - a) 如果您正在使用"代理程序"向导,请单击下一步。
  - b) 单击完成, 以保存该数据源并打开 Agent Editor。

# JSON 数据的 XML 表示

如果 HTTP 请求返回 JSON 数据,那么数据提供程序会将该数据转换为 XML。

数据提供程序会将 JSON 属性名称转换为元素名称。对于简单类型的 JSON 属性,它会将值转换为元素中的 文本数据。嵌入式 JSON 对象会转换为嵌入式 XML 元素。任何下级属性都会转换为下级元素。

根 XML 元素是 JSON\_document。

如果 JSON 属性名称包含不适用于元素名称的字符,那么数据提供程序会进行修改,以生成有效的元素名称。数据提供程序还会向该元素添加一个 JSON\_name 属性。该属性的值是原始 JSON 属性名称。

针对 JSON 数组的每个元素,数据提供程序会创建一个 JSON\_*xxx*\_array\_element XML 元素,其中 *xxx* 是数组名称。该数组元素的值会转换为 XML 元素中的文本。系统会向每个 XML 元素添加一个 JSON\_index 属性;该属性的值是数组中数组元素的索引。

数据提供程序会向每个元素添加下列属性:

·JSON\_level: 节点在 JSON 文件中所处的级别。由 JSON\_document 标记所代表的树根处于级别 1。 ·JSON\_type: JSON 节点的类型(对象、数组、字符串或数字)。

# SOAP 属性的特定字段

在"属性信息"窗口中,有两个 SOAP 属性字段,用于定义如何从 SOAP 响应收集数据。

**属性类型**字段可以是列表中的任意值,用于控制所返回的响应相关信息。某些属性类型要求**类型值**字段包含 值。缺省属性类型为 XPath Query,它对 SOAP 服务器响应内容运行 XPath 查询。此类型值是所运行的 XPath 查询。下表描述所有属性类型及类型值(如果需要):

表 21. SOAP 属性信息						
属性类型	描述	类型值	返回的数据类型	与 FTP 和文件协议 的差别		
XPath Query	对通过 URL 连接返 回的内容运行 XPath 查询。此查 询必须编写为返回 对属性有用的数 据,而不是节点列 表。	要对通过 URL 连接 获取的内容运行的 XPath 查询。如果 定义了行选择查 询,那么此 XPath 查询必须相对于行 选择查询。	返回的数据可以是 字符串、数字或时 间戳记值。用于 SOAP的Agent Builder浏览器通常 从所浏览的数据检 测属性的正确数据 类型。如果数据采 用XML日期时间格 式,那么您可以将 时氟型记指定为属 性类型,代理程序 会将该值转换为 Candle 时间戳记。	无		
Response Time	从所请求的 URL 下 载内容所耗用的时 间(毫秒)。	无	整数(毫秒数)	无		
Response Message	服务器返回的 HTTP 响应消息。	无	字符串	仅当此 URL 使用 HTTP 或 HTTPS 协 议时,响应消息才 适用。		
Response Code	服务器返回的 HTTP 响应代码。	无	整数	仅当此 URL 使用 HTTP 或 HTTPS 协 议时,响应代码才 适用。对于文件或 FTP URL,此代码 始终为 0。		
Response Length	从所请求的 URL 下 载的内容的大小 (以字节计)	无	整数(以字节计的 大小)	无		
Response Header	可用于从其中一个 URL响应头字段检 索值的响应头。自 变量指定所要请求 的字段。	要收集的响应头字段。	字符串	通常,FTP和文件 协议没有任何可收 集的头。		
表 21. SOAP 属性信息 (续)						
---------------------	--	-----	---------	--------------------		
属性类型	描述	类型值	返回的数据类型	与 FTP 和文件协议 的差别		
Request URL	已建立与此 URL 的 连接。所有响应关 键字都将提供有关 与此 URL 的连接的 信息。您可以使用 XPath 查询从通过 访问此 URL 返回的 内容中获取信息。	无	字符串	无		

#### XPath 选项

通过使用 XML 路径语言,您可以从 XML 文档选择节点。SOAP 数据源的 XPath 的一些可能用法包括:

·在 XPath 中使用谓词来标识对应于 IBM Tivoli Monitoring 属性组中的数据行的 XML 元素。您可以在用于将 XML 元素或属性映射至 Tivoli Monitoring 属性的 XPath 中使用谓词,如以下示例所示:

Stat[@name="URLs"]/CountStatistic[@name="URIRequestCount"]/@count

其中,XPath 包含多个位置步骤,每个位置步骤可以包含一个或多个谓词。这些谓词可能比较复杂并包含 布尔值或公式运算符。例如:

//PerformanceMonitor/Node/Server[@name="server1"]/Stat/Stat/Stat[@name= "Servlets"]/Stat

· 在 XPath 中包括节点集函数,前提是某一行包含多个同一类型的 XML 元素。并且节点列表中 XML 元素的 位置确定了此元素所映射到的 Tivoli Monitoring 属性。节点集函数的示例包括 position()、first()、 last()和 count()。

·执行简单数据变换,例如 substring。如果指定以下 substring:

substring(myXMLElement,1,3)

此 XPath 将返回 XML 元素 myXMLElement 的前三个字符。

您可以使用两个句点(..在 Row Selection XPath 上下文外部指定元素,如以下示例所示:

/../OrganizationDescription/OrganizationIdentifier

## SOAP 配置

添加 SOAP 数据源之后, 配置将显示在 Agent Editor 的"运行时配置"页面上。

对于 HTTP 服务器、代理服务器和 Java,添加了相应的配置节。有关代理服务器配置的信息,请参阅(<u>第</u>122页的『代理服务器配置』)。有关 Java 配置的信息,请参阅 <u>第 122 页的『Java 配置』</u>。

#### HTTP 服务器

"HTTP 服务器"配置部分包含以下属性:

表 22. HTTP 服务器配置属性			
姓名	有效值	是否必需	描述
HTTP 用户名	字符串	否	HTTP 用户
HTTP 密码	密码	否	HTTP 服务器密码

表 22. HTTP 服务器配置属性 (续)			
姓名	有效值	是否必需	描述
HTTP 服务器名称	字符串 (缺省值为 localhost)	否	HTTP 服务器的主机或 IP 地址
HTTP 端口号	数字 (缺省值为 80)	否	HTTP 服务器的主机或 IP 地址
启用证书验证	True 或 False (缺省值为 True)	是	禁用证书验证可能不安全
HTTP 信任库文件	文件的路径	否	HTTP 信任库文件
HTTP 信任库密码	HTTP 信任库密码	否	HTTP 信任库密码

#### 代理服务器

如果正在运行代理程序的系统需要代理才能访问 SOAP 数据提供程序,那么必须指定代理服务器配置属性。 有关更多信息,请参阅<u>第122页的『代理服务器配置』</u>。

## 测试 SOAP 属性组

您可以对在 Agent Builder 中创建的 SOAP 属性组进行测试

#### 过程

- 1. 可以通过以下方法启动测试过程:
  - · 在创建代理程序期间,单击 "SOAP 信息"页面上的测试。
  - · 在创建代理程序之后,在 Agent Editor 的"数据源定义"页面上选择属性组,然后单击测试。有关 Agent Editor 的更多信息,请参阅第 17 页的『第 4 章 使用 Agent Editor 修改代理程序』

在上述两个步骤中的某个步骤中单击测试之后,将显示"测试 SOAP 收集"窗口。

2. 可选: 在开始测试之前, 您可以设置环境变量、配置属性和 Java 信息。

有关更多信息,请参阅<u>第191页的『属性组测试』</u>。有关 SOAP 配置的更多信息,请参阅 <u>第129页的</u> <u>『SOAP 配置』</u>。

- 3. 更改 URL、Row Selection XPath 和请求类型。
- 4. 单击启动代理程序。

此时将显示一个窗口,表明代理程序正在启动。

5. 要模拟来自 Tivoli Enterprise Portal 或 SOAP 的代理程序数据请求,请单击**收集数据**。此操作将填充 Results 表,您可以预览数据在 Tivoli Enterprise Portal 中的列中的解析和显示方式。

在 Results 区域中,可以更改属性定义,然后重新装入数据以查看这些更改如何影响属性组。您可以在列结果区域中右键单击以显示用于编辑属性的选项。属性编辑选项为:

- ·编辑属性
- ·隐藏属性
- ·在前面插入属性
- ·在后面插入属性
- ·移除
- ·移除后续属性
- ·全部移除
- 6. 可选: 如果返回的数据未如预期,请单击检查结果。

"数据收集状态"窗口将会打开,并显示有关该数据的更多信息。<u>第 239 页的『Performance Object</u> Status 节点』中描述了"数据收集状态"窗口收集和显示的数据。

7. 通过单击停止代理程序停止代理程序。

8. 单击确定或取消以退出"测试 SOAP 收集"窗口。单击确定将保存您所作的全部更改。

#### 相关概念

<u>第 191 页的『在 Agent Builder 中测试代理程序』</u> 使用 Agent Builder 创建代理程序之后,可以在 Agent Builder 中测试该代理程序。

## 使用套接字来监视数据

您可以定义数据源以使用 TCP 套接字从外部应用程序收集数据。应用程序必须启动与代理程序的 TCP 连接,并以结构化 XML 格式发送数据。根据应用程序,数据源可能产生包含单个行、多行或事件数据的数据集。

#### 关于此任务

使用套接字数据源从外部应用程序向代理程序提供数据,该外部应用程序与代理程序运行相同的系统。这个 外部应用程序可以根据需要随时将数据发送到代理程序。例如,您可以开发一个命令行界面,以便允许用户 在该界面运行时将数据发布到某个属性组。另一种方法是修改受监视的应用程序,使其将更新发送到代理程 序。代理程序不会启动或停止正在将数据发送到套接字的应用程序;该操作由用户进行控制。

以下是一些使用套接字数据源的限制:

- ·缺省情况下,只能连接到本地主机(127.0.0.1)。有关配置代理程序以接受来自远程主机的连接的更多信息,请参阅第137页的『远程套接字端口连接』。
- · 套接字 API 未提供使客户机能够确定哪些子节点可用的机制。客户机可以针对特定子节点发送数据,但它必须已经知道该子节点的名称。

您可以使用以下过程来创建属性组,以便通过传输控制协议 (TCP) 套接字收集数据。

#### 过程

- 1. 在"代理程序初始数据源"页面或"数据源位置"页面中,单击监视数据类别区域中的定制程序。
- 2. 在数据源区域中,单击套接字。
- 3. 单击下一步。
- 4. 在"套接字信息"页面上, 输入属性组名称。
- 5. 输入此属性组的帮助文本。
- 6. 选择这个属性组是生成单一数据行、可生成多个数据行还是生成事件。有关更多信息,请参阅<u>第133页</u>的『发送数据』。
- 7. 在"套接字信息"部分中,选择代码页。有关更多信息,请参阅第135页的『字符集』。
- 8. 可选:单击高级,以修改此属性组的高级属性。当您选择属性组**可生成多个数据行或生成事件**时,高级 选项处于活动状态。
- 9. 单击下一步。
- **10**. 在"特性信息"页面上,指定此属性组的第一个特性。有关创建特性的更多信息,请参阅<u>第 32 页的</u> <u>『创建属性』</u>。
- 11. 单击下一步。
- 12. 可选: 在 "全局套接字数据源信息"页面上的错误码部分中,定义套接字客户机在无法收集数据时可以 发送的错误代码。有关更多信息,请参阅<u>第134页的『发送错误而不是数据』</u>。要定义错误代码,请使 用以下步骤:
  - a) 在错误码部分中, 单击添加。错误代码限长 256 个字符。只允许使用 ASCII 字母、数字和下划线。 不允许使用空格。
  - b) 在"套接字错误码定义"窗口中, 输入 Performance Object Status 属性组中显示的值。
  - c) 输入内部值。内部值必须是介于 1,000 与 2,147,483,647 之间的整数。

- d) 必须为每个错误定义消息文本。您可以通过从列表中进行选择来使用先前输入的消息文本。单击确定,以返回到"全局套接字数据源信息"页面。该消息文本在代理程序日志文件中使用。如果没有合适的可用消息文本,请单击浏览以设置消息文本。此时将打开 Messages (列表)窗口。Messages 窗口列出代理程序中定义的消息。在定义消息之前,该列表保持空白。您可以使用编辑来变更已定义的消息,并可以使用移除来删除一条或多条已定义的消息。
- e) 在"消息"(列表)窗口中,单击**添加**以查看"**消息定义**"窗口。在"**消息定义**"窗口中,输入用 于描述新消息含义的文本,并选择消息类型。

注:系统会为您自动生成消息标识。

- f) 单击确定。
- g) 此时将打开 Messages (列表) 窗口, 其中显示了新消息。要验证消息并返回到"**全局套接字数据源** 信息"页面, 请单击确定。
- **13**. 可选: 在"**全局套接字数据源信息**"页面的**补充文件**部分中,您可以添加与代理程序打包在一起的文件。这些文件会在安装该代理程序时复制到代理程序系统。

**文件类型**列描述每个文件的预期用法。下表描述了三种可能的用途:

表 23. 补充文件的文件类型		
文件类型	描述	
Executable	如果要随代理程序一起提供可执行文件,请选择此 选项。代理程序不使用这些文件。	
Library	如果要随代理程序一起提供库,请选择此选项。代 理程序不使用这些文件。	
Java resource	如果要随代理程序一起提供 Java 资源,请选择此选项。代理程序不使用这些文件。	

有关随代理程序一起安装的补充文件安装位置的信息,请参阅(第 206 页的『系统上的新文件』)。

单击编辑可以对导入的文件进行编辑。有关更多信息,请参阅(<u>第104页的『编辑命令文件定义』</u>)。

- 14. 可选: 您可以通过单击**测试**来测试此属性组。有关测试的更多信息,请参阅<u>第138页的『测试套接字</u> 属性组』
- 15. 可选:如果已对数据源采样,您可以单击高级来创建过滤器以限制此属性组返回的数据。如果没有在 "套接字信息"页面上选择"生成事件",那么会对数据源采样。有关过滤来自属性组的数据的更多信息,请参阅<u>第 38 页的『过滤属性组』</u>
- 16. 请执行以下步骤之一:

a) 如果您正在使用"代理程序"向导,请单击下一步。

中的任何套接字属性组页面上单击全局设置。

b) 单击完成,以保存该数据源并打开 Agent Editor。 在"套接字提供程序设置"页面的操作系统部分中,选择代理程序侦听来自套接字客户机的数据时 所在的操作系统。要打开此页面,请在大纲视图中单击套接字提供程序设置,或者在 Agent Editor

注:可以在"套接字提供程序设置"页面的错误码和补充文件部分中更新错误码及补充文件。

#### 将套接字信息发送到代理程序

当代理程序包含一个或多个套接字属性组时,该代理程序将打开一个套接字并侦听来自客户机的数据。

将套接字数据发送到代理程序的应用程序连接到该代理程序中已定义的端口。该端口是代理程序配置属性设置的值,或者是按照 TCP/IP 自动分配的临时端口。有关套接字端口和配置的更多信息,请参阅<u>第 136 页的</u> <u>『套接字配置』</u>。

所接收的数据必须遵循结构化 XML 格式。使用套接字数据源可以实现以下 XML 信息流:

- ·对于采样属性组,将一个或多个数据行发送到代理程序
- ·对于生成事件的属性组,将一个数据行发送到代理程序

- ·将错误代码发送到代理程序,而非数据。
- ·将任务前缀注册发送到代理程序
- ·接收来自代理程序的任务请求
- ·将任务响应发送到代理程序

#### 发送数据

定义属性组以接收采样数据或事件数据。在创建属性组时,您将指定一个选项以指示要接收的数据应执行以 下哪种操作:

- ·生成单一数据行
- ·生成多个数据行
- ·生成事件

如果您选择**生成单一数据行**或**可生成多个数据行**,则为采样属性组。如果选择**生成事件**,那么每次收到行时,属性组都会向监视环境发送事件。

在 Tivoli Enterprise Portal 或 IBM Cloud Application Performance Management 控制台中查看采样数据时, 会看到最新收集的一组数据行。针对事件属性组显示的数据是代理程序维护的本地高速缓存的内容。对于事 件数据,代理程序会向高速缓存添加新条目,直到达到此大小为止,此时会删除最旧的条目。对于采样数 据,您每次发送数据时,代理程序都将替换高速缓存的内容。

如果选择了**生成事件**或**生成单一数据行**,则必须在每条消息中针对该属性组,向代理程序仅发送一行数据。 您可以根据需要发送任意数目的事件,并在独立的消息中发送各个事件。

通常,采样数据按请求由代理程序收集,但套接字客户机将按自己的时间表提供已更新的样本。您可以根据 需要按任意频率更新采样属性组(单行或多行)。在 Tivoli Monitoring 或 IBM Cloud Application Performance Management 请求数据时,代理程序会提供最新的数据。

如果 Tivoli Enterprise Portal 或 IBM Cloud Application Performance Management 控制台中缺少套接字属性 组的数据行,请检查日志文件中是否存在错误。此外,如果该属性组中的数据与期望不符,也请检查日志文 件中是否存在错误。套接字数据源将尝试处理它所能够处理的任何输入内容。例如,如果客户机发送了三个 格式正确的行和一个无效的行(例如,格式不正确的 XML),那么您将看到:

·在属性组中有三个数据行

·在代理程序的日志文件中,针对格式不正确的行记录了错误

·由于返回了有效的行,因此 Performance Object Status 将显示 NO\_ERROR 状态

对于事件数据和采样数据,数据作为单一 XML 数据流从套接字客户机发送到代理程序。从套接字客户机发送 的数据必须始终以换行符"\n"结尾。代理程序将读取换行符之前的所有数据,然后尝试处理接收到的数 据。接收到的任何无法处理的数据都将被废弃。以下样本说明如何将名为 abc 的属性组的两个数据行发送到 代理程序:

<socketData><attrGroup name="abc"><in><a v="1"/><a v="no"/><a v="5"/></in><in> \ <a v="3"/><a v="yes"/><a v="5"/></in></attrGroup></socketData>\n

此样本将两个数据行发送到代理程序,并且每个数据行都包含三个属性。这些属性的顺序很重要,并且必须 遵循属性组中定义的顺序。唯一的例外情况是,必须跳过派生的属性,而不必考虑这些属性在属性组中的位置。

如果在子节点中定义此属性组,那么向代理程序发送数据时,必须标识子节点实例标识。通过使用 socketData 元素中的子节点属性来标识子节点实例标识。在配置供套接字客户机使用的子节点实例标识时 必须采用一种约定,因为客户机无法查询实例标识或配置属性。发送到未配置的子节点的数据将被忽略。

以下是一个样本:

<socketData subnode="app1"><attrGroup name="abc"><in><a v="1"/><a v="no"/><a v="5"/>
</in><in> \<a v="3"/><a v="yes"/><a v="5"/></in></attrGroup></socketData>\n

在此示例中,将数据发送到实例标识为 "app1" 的子节点。"app1" 不是受管系统名称,而是在配置子节点 实例时指定的实例标识。

套接字数据由下列 XML 元素构成:

#### socketData

根元素。它包含一个可选属性,名为 subnode,用于指定子节点实例标识。

#### attrGroup

此元素用于标识套接字数据的相应属性组。name 属性是必需的,用于指定属性组名。

in a

a 元素用于标识属性值。v 属性是必需的, 用于指定属性值。

#### 发送错误而不是数据

有时,发布套接字数据的应用程序可能无法收集属性组所必需的数据。在这种情况下,可以返回错误代码, 而不是向代理程序发送数据。错误代码为您提供了如何将问题告知监视环境的方法。以下是一个错误示例:

此元素是必需的,用于标识新的数据行。一个数据行的所有属性值都必须是同一个 in 元素的子代。

<socketData><attrGroup name="abc"/><error rc="1000"/></attrGroup></socketData>\n

必须在代理程序中的一个所有套接字属性组公用的列表中定义错误代码。当代理程序收到错误代码时,将在代理程序日志文件中记录定义的错误消息。另外,名为 Performance Object Status 的属性组还包含一个

"错误代码"属性,此属性将通过"错误代码类型"更新。该"错误代码类型"是针对发送的错误代码定义的。

对于上述示例,您必须已在代理程序中定义错误代码值1000。请参阅以下错误代码定义样本:

表 24. 样本错误代码

错误代码值	错误代码类型	消息
1000	APP_NOT_RUNNING	The application is not running

发送此错误代码时,将在代理程序日志文件中记录一条类似于以下示例的消息:

(4D7FA153.0000-5:customproviderserver.cpp,1799,"processRC") Received error code 1000 from client. \Message: K1C0001E The application is not running

如果从 Tivoli Enterprise Portal 选择了 Performance Object Status 查询,那么行 **abc** 属性组的错误码列将显示该表中的值 APP\_NOT\_RUNNING。

向采样属性组发送错误将清除先前针对该属性组收到的任何数据。向该属性组发送数据将导致此错误代码不再显示在 Performance Object Status 属性组中。另外,您也可以发送错误代码 0 以清除该表中的此错误代码。

向产生事件的属性组发送错误并不会从高速缓存中清除先前发送的事件。

#### 处理"执行操作"请求

套接字客户机可以进行注册,以便在操作命令与特定前缀相匹配时接收来自代理程序的"执行操作"请求。 任何不匹配的操作都由代理程序处理。该前缀不得与代理程序应该处理的操作相冲突,因此,请使用代理程 序产品代码作为前缀。随 Agent Builder 一起提供的"执行操作"根据"执行操作"使用的数据源进行命 名。例如,"执行操作"JMX\_INVOKE 对 JMX 数据源进行操作。另一个示例是使用 SSH 脚本数据提供程序 的"执行操作"SSHEXEC。由于这些操作不使用产品代码,因此,可以安全地将产品代码用作"执行操作" 前缀。

套接字客户机必须长时间运行,并且必须保持套接字处于打开状态。它必须针对前缀发送注册请求,并侦听 来自套接字的请求。即使数据未在流动,代理程序也能确保不会发生长时间运行客户机的套接字超时。以下 是注册请求样本:

<taskPrefix value="K42"/>\n

在此样本中,代理程序收到的并以 "K42" 开头的任何"执行操作"命令将转发到发起注册的套接字客户机。以下样本显示套接字客户机可能收到的"执行操作"请求:

<taskRequest id="1"><task command="K42 refresh" user="sysadmin"/></taskRequest>\n

id 是一个唯一标识,代理程序用来跟踪发送到客户机的请求。当套接字客户机响应该任务时,它必须在 taskResponse 元素的 id 属性中提供此标识。

套接字客户机必须处理操作并发送响应。以下是一个响应样本:

<taskResponse id="1" rc="1"/>\n

如果操作成功完成,那么将返回 rc 属性值 0。rc 的值必须是整数,其中,任何非 0 值都将被视为失败。任务返回码值将记录到代理程序日志文件中,并显示在代理程序所包含的 Take Action Status 查询中。运行操作后在 Tivoli Enterprise Portal 上显示的对话框没有显示返回码。该对话框将指示"执行操作"命令是返回了成功还是失败。如果发生失败,您必须查看代理程序日志或 Take Action Status 查询以确定实际返回码。

代理程序开发人员负责记录、创建和导入与代理程序一起使用的套接字客户机所支持的任何操作。如果用户 向套接字客户机发送了不受支持的操作,那么必须将客户机开发为能够以适当的方式处理这些场景。如果用 户定义了更多以注册前缀开头的更多操作,这些操作将传递给客户机。必须将客户机开发为能够以适当的方 式处理这些场景。

代理程序在等待来自套接字客户机的响应时,等待时间由超时控制。该设置是在代理程序中定义的环境变量,名为 CDP\_DP\_ACTION\_TIMEOUT,其缺省值是 20 秒。

**注:** 为套接字数据源属性组定义的错误代码消息不用于"执行操作"。您可以返回相同的返回码值。但是, 代理程序不会记录定义的消息,也不会影响 Performance Object Status 属性组中的错误代码字段。

#### 套接字数据的编码

套接字客户机对发送到代理程序的数据进行编码。

了解套接字客户机对要发送至代理程序的数据进行编码的方式非常重要。

#### 特殊字符

发送到代理程序的数据不得包含任何换行符(在每个事件或数据样本的末尾除外)。属性值中出现的换行符 必须替换为其他字符或(<u>第 135 页的表 25</u>)中编码的字符。另外,您还必须确保属性值不违反 XML 语法。 下表显示属性值中进行编码的字符:

表 25. 属性值中要进行编码的字符		
字符	头	
&	&	
<	<	
>	>	
"	"	
1	'	
\n		

注:代理程序使用换行符来分隔从客户机接收到的响应。意外出现的换行符导致无法正确地解析数据。

代理程序不包含全功能型 XML 解析器,因此您不得对不在(<u>第135页的表25</u>)中的字符使用特殊编码。例如,不要编码 &#162;或 &cent 以代替分币符号 ¢。

#### 字符集

除了要对特殊字符编码,代理程序还必须了解用于对数据进行编码的代码页。在定义每个套接字属性组时, 应指出您是要将数据作为 UTF-8 数据还是本地代码页数据发送到代理程序。需了解客户机发送数据的方式。 如果使用以 Java 编写的客户机,请在写程序中指定 UTF-8 编码以用于向代理程序发送数据。请指定 UTF-8 作为属性组的代码页。本地代码页表示代理程序的本地代码页。如果通过远程套接字来发送数据,那么必须 遵循代理程序的本地代码页或者使用 UTF-8。

#### 数字数据

应了解您的数字属性值的格式设置。发送到代理程序的数字值不得包含任何特殊字符。千位分隔符就是特殊 字符的一个示例。其他示例包括货币符号或者用于描述值单位的字符。如果代理程序在解析数字数据时遇到 问题,将记录错误以指示该问题。解析属性失败时,不会设置 Performance Object Status 错误代码。以下 是代理程序日志中的错误消息示例:

(4D3F1FD6.0021-9:utilities.cpp,205,"parseNumericString") Invalid characters :00:04 \ found getting numeric value from 00:00:04, returning 0.000000

注: 有关时间戳记属性必须设置为何种格式的信息, 请参阅(第36页的『时间戳记』)。

#### 套接字错误

对于从套接字客户机接收到的数据所发生的问题,系统会将对应错误写入代理程序日志文件。

记录的其他错误是返回非0值的执行操作。套接字客户机发送的错误值将随着错误代码的关联消息一起记录。

当套接字客户机向代理程序发送错误返回码时,会将属性组设置为 Performance Object Status。除代理程序 定义的值以外,还会看到一些其他值。下表描述了您使用套接字属性组时可能会遇到的其他"错误代码" 值:

表 26. Performance Object Status 值		
错误代码	描述	
NO_ERROR	未发生错误。表示属性组没有任何问题。一行采样数 据出现问题并不会导致状态由 NO_ERROR 更改为其 他值。即使看到错误代码 NO_ERROR,也必须验证 显示的行数和属性值。	
NO_INSTANCES_RETURNED	对于采样属性组, 套接字客户机未发送数据行。并不 是错误。这表示此属性组未监视任何资源实例。	
XML_PARSE_ERROR	代理程序未能解析从客户机收到的数据。有关更多详 细信息,请参阅代理程序日志。	
OBJECT_CURRENTLY_UNAVAILABLE	客户机向代理程序发送了在错误代码的全局列表中未 定义的错误代码。	
GENERAL_ERROR	从客户机收集数据时发生问题,这通常是因为客户机 未在超时时间间隔内应答请求。有关更多详细信息, 请参阅代理程序跟踪日志。	
	另外,客户机也可以指定 GENERAL_ERROR 作为错误代码,但最好定义更详细的错误代码。	

## 套接字配置

向代理程序添加套接字数据源之后,可以将代理程序配置为接受来自指定套接字端口的数据。

#### 关于此任务

添加套接字数据源之后,配置将显示在 Agent Editor 的"运行时配置"页面上。套接字配置节包含以下属性:

表 27. 套接字配置属性			
姓名	有效值	是否必需	描述
端口号	0 或任何正整数 缺省值为 0	是	这是代理程序用于侦听来 自套接字客户机的数据的 端口。值为0表示使用临 时端口。

代理程序会将正在使用的端口值写入一个文件。以后,在代理程序计算机上运行的套接字客户机可以读取此 文件以确定所要连接的端口。端口所写入的文件命名为 kxx\_instanceName\_cps.properties,其中 kxx 是代理程序的三字符产品代码, instanceName 是多实例代理程序的代理程序实例名。如果此代理程序不 是多实例代理程序,那么名称中将不包含这一部分,因此文件名为 kxx\_cp.properties。

在 Windows 中,此文件写入 32 位安装的 %CANDLE\_HOME%\TMAITM6 目录,或者 64 位安装的 %CANDLE\_HOME%\TMAITM6\_x64。在 UNIX 中,此文件写入 /tmp。

#### 过程

- 1. 可选:如果您的系统有多个接口,将环境变量 CDP\_DP\_HOSTNAME 设置为您的网络接口的主机名或 IP 地址:
  - a) 转至 Agent Editor 的"代理程序信息"视图并选择环境变量。
  - b) 单击添加, 然后使用"名称"字段从环境变量列表中选择 CDP\_DP\_HOSTNAME。
  - c) 在 Value 字段中设置主机名或 IP 地址。
- 2. 启动代理程序。

代理程序启动后, 它将与 CDP\_DP\_HOSTNAME 环境变量所定义的接口绑定。如果未设置 CDP\_DP\_HOSTNAME, 那么代理程序将与缺省主机名绑定。

如果要将代理程序绑定到定义的端口,而不是临时端口,可以设置配置属性端口号 (CP\_PORT)。

要设置端口号配置属性,请使用以下步骤:

- a) 转至 Agent Editor 的"运行时配置"视图。
- b) 在"运行时配置信息"窗格中,选择套接字配置 > 套接字 > 端口号。
- c) 在 **Default value** 中输入端口号值。 如果不输入值,那么将使用值 0。值为 0 表示使用临时端口。

#### 远程套接字端口连接

您可以将代理程序配置为接受来自远程套接字端口的数据。代理程序必须在一个与远程系统建立了网络接口连接的系统上运行。

#### 过程

- 1. 通过完成以下步骤将环境变量 CDP\_DP\_ALLOW\_REMOTE 的值设置为 YES。
  - a) 转至 Agent Editor 的"代理程序信息"页面并选择环境变量。
  - b) 单击添加, 然后使用"名称"字段从环境变量列表中选择 CDP\_DP\_ALLOW\_REMOTE。
  - c) 将 Value 字段设置为 YES。
- 2. 遵循 第 136 页的『套接字配置』中详细描述的过程。

限制:

- ·套接字应用程序和代理程序之间发送的数据:
- 必须符合为套接字数据提供程序定义的 XML 语法。有关更多信息,请参阅<u>第135页的『套接字数据</u>的编码』。
- 必须以 UTF-8 编码。

- 明文(未加密)发送。如果此数据包含敏感信息,必须在代理程序外部通过 SSH 通道或其他机制来保护通信安全。

·代理程序处理从任何远程主机收到的数据,因此必须使用适当的防火墙或网络流量过滤器保护环境安全。

#### 结果

您可以在能够连接到代理程序运行所在系统的任何系统上运行实施套接字数据提供程序的代码。

#### 套接字的样本脚本

此样本脚本演示如何编写套接字客户机。

#### Perl 样本

以下样本 Perl 脚本连接至套接字并发送数据。此样本针对在 UNIX 上运行的代理程序而编写, 使用的产品代码为 k00, 并且有一个名为 SocketData 的属性组。

```
#!/usr/bin/perl -w
# SocketTest.pl
# A simple Agent Builder Socket client using IO:Socket
use strict;
use IO:::Socket;
# Initialize socket connection to the agent
#--
my $host = '127.0.0.1';
my port = 0;
if (/^CP_PORT=([0-9]+)/) {
         $port = $1;
    }
}
if ($port == 0) {
    die "Could not find port to use to connect to agent.\n";
ş
my $sock = new IO::Socket::INET( PeerAddr => $host, PeerPort => $port,
Proto => 'tcp'); $sock or die "no socket :$!";
# The following call sends 2 rows of data to the agent. Each row contains 1
# String attribute and 3 numeric attributes.
syswrite $sock, "<socketData><attrGroup name=\"SocketData\"><in><a v=\"A message
from perl\"/> \<a v=\"1\"/><a v=\"2\"/><a v=\"123\"/></in><in><a v=\"More from
perl\"/><a v=\"456\"/> \<a v=\"123\"/><a v=\"789\"/></in></attrGroup>
</socketData>\n";
close $sock;
```

## 测试套接字属性组

您可以对在 Agent Builder 中创建的套接字属性组进行测试。

#### 开始之前

要测试此属性组,您需要一个用于发送数据的套接字客户机。可以在<u>第138页的『套接字的样本脚本』</u>中查 看使用 Perl 脚本编写的示例套接字客户机

**限制:** 与大多数其他属性组不同的是, 您无法在创建期间测试套接字属性组。您可以在完成此属性组的创建 工作后对其进行测试。

#### 过程

1. 创建代理程序之后,请在 Agent Editor 的"数据源定义"页面上选择属性组,然后单击测试。有关 Agent Editor 的更多信息,请参阅<u>第 17 页的『第 4 章 使用 Agent Editor 修改代理程序』</u>。

在上述两个步骤中的某个步骤中单击测试之后,将显示"测试套接字客户机"窗口。

- 可选: 在开始测试之前设置环境变量和配置属性。
   有关更多信息,请参阅<u>第191页的『属性组测试』</u>。
- 3. 单击启动代理程序。此时将显示一个窗口, 表明代理程序正在启动。
- 4. 当代理程序启动时, 会根据其配置侦听套接字数据。
- 5. 要测试代理程序的数据收集,您现在应该生成与代理程序配置匹配的套接字数据。 您可以使用套接字客户机生成套接字数据。 当代理程序接收到与其配置匹配的套接字数据时,会将这些数据添加到其内部高速缓存。
- 6. 要模拟来自 Tivoli Enterprise Portal 的代理程序数据请求,请单击**收集数据**。

"测试套接字客户机"窗口会收集并显示自代理程序上次启动以来,代理程序高速缓存中的全部数据。

7. 如果某些操作似乎未按预期工作,请单击检查结果。

"数据收集状态"窗口将会打开,并显示有关该数据的更多信息。<u>第 239 页的『Performance Object</u> <u>Status 节点』</u>中描述了 Data Collection Status 窗口收集和显示的数据

8. 通过单击停止代理程序停止代理程序。

9. 单击确定或取消以退出"测试套接字客户机"窗口。单击确定将保存您所作的全部更改。

#### 相关概念

<u>第191页的『在Agent Builder 中测试代理程序』</u> 使用 Agent Builder 创建代理程序之后,可以在 Agent Builder 中测试该代理程序。

## 使用 Java API 来监视数据

您可以定义数据源以使用 Java API 与 Java 平台上长期运行的应用程序交互。代理程序在启动时启动应用程序并定期与其进行交互。在构建代理程序时, Agent Builder 创建应用程序的源代码。您必须定制代码以收集正确的数据。根据代码,数据源可以生成多个数据集,可包含一行、多行或事件数据。

#### 关于此任务

使用 Java API 数据源和 Java 编程语言来收集使用其他 Agent Builder 数据源无法收集的数据。代理程序将 启动此 Java 应用程序并在到时间关闭时发送关闭请求。此 Java 应用程序只能在接收到退出请求时退出。

包含 Java API 属性组的代理程序会与 Java 应用程序进程进行交互。Java 应用程序使用 Java 提供程序客户 机 API 与代理程序进行交互。有关此 API 的信息,请参阅 Tivoli Monitoring Knowledge Center 上的 <u>Javadoc</u>。您可以使用 Java API 来执行下列操作:

- ·连接到代理程序进程,并针对 Java 应用程序所支持的属性组进行注册
- ·接收并应答对采样的数据发出的请求
- ·以异步方式为生成事件的属性组发送数据
- ·对于收集数据失败的属性组,发送错误
- ·在配置了子节点实例的子节点中支持属性组
- ·接收并回复"执行操作"请求

请使用以下过程来创建在 Java 应用程序中收集数据并使用 Java API 发送数据的属性组。该过程显示如何创建样本 Java 应用程序作为您自己的 Java 应用程序的基础。

#### 过程

1. 在"代理程序初始数据源"页面或"数据源位置"页面中,单击监视数据类别区域中的定制程序。

- 2. 在**数据源**区域中,单击 **Java API**。
- 3. 单击下一步。

- 4. 在 Java API 信息页面上, 输入属性组名称。
- 5. 输入此属性组的帮助文本。
- 6. 选择这个属性组是生成单一数据行、可生成多个数据行还是生成事件。此选项将影响在向导末尾创建的 样本 Java 应用程序。有关更多信息,请参阅<u>第 133 页的『发送数据』</u>。
- 7. 可选: 单击高级,以修改此属性组的高级属性。当您选择属性组**可生成多个数据行或生成事件**时,高级可用。
- 8. 单击下一步。
- 9. 在"特性信息"页面上,指定此属性组的第一个特性。有关创建特性的更多信息,请参阅(<u>第32页的</u> <u>『创建属性』</u>)。
- 10. 选择**添加其他特性**,并单击**下一步**以便向代理程序添加其他特性。对特性的引用将合并到在向导结束时 创建的样本 Java 应用程序中。
- 11. 单击下一步。
- 12. 在全局 Java API 数据源信息页面上, 输入类名及 JAR 文件名。

这是一个标准类名,在 Java 启动时,将调用这个类的 main 方法。将使用此类中的主 Java 方法创建样本 Java 应用程序。

此 JAR 文件是一个归档, 它包含构成该 Java 应用程序的 Java 类。此 JAR 文件将随代理程序一起打包和安装。

13. 可选: 在"**全局 Java API 数据源信息**"页面上的**错误码**部分中,定义 Java 应用程序可以发送的错误码。这些错误码由 Java 应用程序在其无法收集数据时发送。

限制:错误代码限长 256 个字符。只允许使用 ASCII 字母、数字和下划线。不允许使用空格。

- a) 单击"错误码"部分中的添加。
- b) 在 Java API 错误码定义窗口中, 输入显示值。
- c) 输入内部值。内部值必须是介于 1,000 与 2,147,483,647 之间的整数。
- d) 定义每个错误的消息文本。您可以通过从列表中进行选择来使用先前输入的消息文本。单击确定, 以返回到全局 Java API 数据源信息页面。 此消息将记录在代理程序日志文件中。
- e) 如果没有合适的可用消息文本,请单击浏览以设置消息文本。

这将显示"消息"(列表)窗口。"消息"窗口列出代理程序中定义的消息。在定义消息之前,列 表保留为空。您可以使用**编辑**来变更已定义的消息,并可以使用**移除**来删除一条或多条已定义的消 息。

f) 在"消息"(列表)窗口中, 单击**添加**以查看"**消息定义**"窗口。在"**消息定义**"窗口中, 您可以 输入文本以描述新消息的含义并选择消息类型。

注:系统会为您自动生成消息标识。

- g) 单击确定。
- h) 这将显示包含新消息的"消息"(列表)窗口。要验证消息并返回到"全局 Java API 数据源信息" 页面,请单击确定。
- 14. 可选: 在全局 Java API 数据源信息页面的补充文件部分中,可以添加文件,这些文件将与代理程序一起打包并在代理程序安装时复制到代理程序系统。Java 提供程序客户机 API JAR 文件未在此处列出;此文件自动复制到代理程序系统。文件类型列描述每个文件的预期用法。下表(<u>第 140 页的表 28</u>)描述了三种可能的用法。单击编辑可以对导入的文件进行编辑。有关更多信息,请参阅<u>第 104 页的『编辑命令文件定义』</u>。

表 28. 补充文件的文件类型		
文件类型	描述	
Executable	如果要随代理程序一起提供可执行文件,请选择此 选项。代理程序不使用此文件,但此文件将保存在 路径中供 Java 应用程序使用。	

表 28. 补充文件的文件类型 (续)		
文件类型	描述	
Library	如果要随代理程序一起提供库,请选择此选项。代 理程序不使用此文件,但此文件将保存在库路径中 供 Java 应用程序使用。	
Java resource	如果要随代理程序一起提供 Java 资源,请选择此 选项。代理程序不使用此文件,但此文件将保存在 类路径中供 Java 应用程序使用。	

**注:** 当 Java 资源补充文件添加到 Agent Builder 时,该文件将自动添加到项目类路径中。Java 编译器使用这个补充文件将代码中包含的所有引用解析为资源中的类。

有关补充文件随代理程序一起安装于何处的信息,请参阅第 206 页的『系统上的新文件』。

**15.** 可选: 创建过滤器,以限制此属性组所返回的数据(如果对数据进行取样)。通过单击**高级**创建过滤器。

注: 如果在"Java API 信息"页面上未选中生成事件,那么将对数据进行取样。

有关对来自属性组的数据进行过滤的更多信息,请参阅第38页的『过滤属性组』。

16. 可选: 为子节点添加配置属性

如果您正将此数据源添加到子节点,那么将显示"**子节点配置覆盖**"页面,因此您可以向该子节点添加 配置属性。要创建样本 Java 应用程序,在子节点下需要至少一个配置属性。因为样本使用配置属性来 区分不同的子节点实例,所以至少需要一个配置属性。

#### 17. 执行下列其中一个步骤:

a) 如果您正在使用"代理程序"向导,请单击下一步。根据需要完成向导。

- b) 否则,请单击完成,以保存该数据源并打开 Agent Editor。然后,在主菜单中,选择文件 > 保存。
- 此时, Agent Builder 将创建监视应用程序的源代码。此代码位于项目目录的 src 子目录中。请编辑此代码,以创建监视应用程序。

#### 下一步做什么

在"Java API 设置"页面上,选择正确的操作系统。如果要运行此属性组和 Java 应用程序的操作系统不同 于您为代理程序定义的操作系统,请进行此选择。要打开此页面,请在大纲视图中单击 Java API 设置,或 者在 Agent Editor 中的任何 Java API 属性组页面上单击全局设置。

注: 稍后,可以在 Java API 设置页面的错误码和补充文件部分中更新错误码及补充文件。

## 运行 Java 应用程序

有关 Java 应用程序及其依赖关系的初始化的信息

#### 初始化 Java 应用程序

代理程序将在启动和初始化期间启动 Java 应用程序。配置设置用于控制使用哪个 Java 运行时来启动该进程。另外,您还可以在配置中指定 Java 虚拟机参数和 Java 日志记录级别。有关 Java API 配置的更多信息,请参阅 <u>第 149 页的『Java API 配置』</u>。Java 进程将继承为代理程序定义的环境变量。运行时配置设置也被置于环境中,并且,您可以使用 API 调用来查询这些设置。

Java 应用程序必须是长时间运行的进程。除非它接收到来自 API 的关闭请求,否则不能终止。如果 Java 应用程序在向代理程序注册后终止,那么代理程序将尝试重新启动该 Java 应用程序,并且最多尝试三次。如果成功地恢复数据收集,那么此重新启动计数将重置。在 Java 应用程序终止时以及在发起重新启动时,代理程序将记录错误。

注:如果 Java 应用程序在属性组注册完成之前终止,那么将不会尝试重新启动。

#### 依赖项

Java 应用程序必须使用 Java 运行时环境。以下 Java 版本受支持:

·Oracle Corporation Java V5 或更高版本

· IBM Corporation Java V5 或更高版本

必须先在代理程序系统上安装 Java,然后再配置并启动代理程序。包含用于与代理程序进行通信的 API 的 JAR 文件将随代理程序运行时一起提供,并且包括在 JVM 的类路径中。必须将 Java 应用程序所需的任何其 他 JAR 文件都定义为 Java API 属性组的补充文件。任何文件类型为 Java 资源的补充文件都将随 Java API JAR 文件一起自动添加到 Java 应用程序的基本类路径中。

任何未随代理程序一起提供但却是 Java 应用程序运行时操作所必需的 JAR 文件都必须包括在外部 JAR 的类路径配置设置中。

## 生成的样本 Java 应用程序

此参考资料描述 Agent Builder 生成的代码以及您必须针对所要监视的资源添加或替换的代码。

当您创建具有一个或多个 Java API 数据源的代理程序时, Agent Builder 将生成 Java 应用程序源代码。该代 码将在代理程序项目中生成并遵循代理程序的结构。您必须向生成的应用程序添加自己的 Java 代码。您的 代码收集采样属性组的数据、处理要向基于事件的属性组发布的事件、在遇到任何问题时报告错误以及运行 任务。生成的应用程序为代理程序提供数据, 但这是样本数据, 将替换为从您要监视的资源获取的数据。

假设存在一个具有下列特征的样本代理程序:

- ·产品代码: K91
- ·Java API 主类: agent.client.MainClass
- · <u>第 142 页的图 17</u> 所示的代理程序数据源结构:
  - Sample Agent using Java API (Agent) Sampled\_Data (Java API) ....123 Int32 123 Int64 "\$" Str DateAndTime Int64Scaled 🖮 📋 Event\_Data (Java API) .....123 Int32 "5" Str 😑 🖓 🍅 Some subnode (Subnode) 🖮 📋 Sampled\_Subnode (Java API) 123 Int32 123 Int64 --- "\$" Str DateAndTime 🖮 🗻 Event\_Subnode (Java API) 123 Int32 123 Int64 "5" Str DateAndTime

图 17. 样本代理程序结构

·某个子节点配置属性: K91\_INSTANCE\_KEY

#### 类结构

在生成的 Java 应用程序中,与代理程序进行交互的代码在很大程度上独立于与您所监视的资源进行交互的代码。它包含您修改的文件以及不修改的文件。

下列 Java 类由 Agent Builder 创建:

#### MainClass(agent.client 包)

您在"**全局 Java API 数据源信息**"页面上指定的类。这个类包含 main 方法以及用于处理执行操作请求 的方法。这个类继承自下面描述的帮助程序类。您必须修改这个类,以便与要监视的资源以及要执行的 操作进行交互。

#### MainClassBase (agent.client 包)

这是一个帮助程序类,用于初始化服务器连接、注册属性组并等待来自服务器的请求。请不要修改这个类。

#### Sampled\_Data、Sampled\_Subnode、Event\_Data 和 Event\_Subnode 类

(agent.client.attributeGroups 包)

每个 Java API 属性组都有一个类,用于处理该属性组的数据收集请求或者为该属性组生成事件。这些类中的每个类都继承自下面描述的其中一个帮助程序类。您必须修改这些类,以便从要监视的资源收集数据。

Sampled\_DataBase、Sampled\_SubnodeBase、Event\_DataBase和Event\_SubnodeBase类 (agent.client.attributeGroups 包)

这些是帮助程序类(每个 Java API 属性组都有一个对应的帮助程序类),用于定义组属性在内部类中的结构。请不要修改这些类。

#### ICustomAttributeGroup 接口(agent.client.attributeGroups 包)

此接口用于定义每个属性组类中的公用方法。请不要修改此接口。

您可以修改的类永远不会被 Agent Builder 覆盖。仅当这些类不存在时, Agent Builder 才会创建这些类。

每次保存 Agent Builder 时,帮助程序类和接口都将被覆盖。对代理程序进行更改并保存该代理程序时,帮助程序类将更新以反映对 Java API 属性组所作的任何结构更改。接口和帮助程序类的头部包含警告,提醒您不要修改该文件。

#### 初始化与清除

代理程序启动时,将调用 MainClass 中的 main 方法。此方法将创建 MainClass 实例,然后进入用于接 收并处理代理程序请求的长时间运行方法。

大部分初始化和清除代码必须添加到 MainClass 中。在构造函数中,请添加创建或访问资源所需的初始化代码。您可能想打开远程资源连接、创建句柄或初始化数据结构。

在代理程序终止之前,将调用 stopDataCollection 方法。如果要在 Java 应用程序结束前关闭连接或执行清除操作,请在 stopDataCollection 方法中添加相应的代码。

如果只需要针对特定属性组进行初始化,那么可以在该属性组类的构造函数中添加初始化代码。同样,如果只需要针对特定属性组执行任何清除操作,那么可以在该属性组的 stopDataCollection 方法中添加相应的清除代码。

Java 应用程序中的任何代码都可以使用记录器对象来写入日志条目。(主帮助程序类将在它的构造函数中创建一个受保护的记录器对象。属性组帮助程序对象将在它们的构造函数中创建对该记录器的受保护引用)。此记录器对象将使用 Java 跟踪日志实用程序。可以从记录器所创建的跟踪日志中获取错误及详细跟踪信息。跟踪信息对于诊断提供程序存在的问题而言非常重要。

在调用 stopDataCollection 时,如果将清除工作传递给另一个线程,请先等待该线程完成,然后再从 stopDataCollection 方法返回。否则,在进程结束时,清除工作可能会由于主线程已完成而突然终止。

在代理程序配置设置中,其中一个是 Java 跟踪级别设置。下表显示可以在 JAVA\_TRACE\_LEVEL 配置属性 中设置的值。如果 API 创建了记录器,那么下表将显示该记录器使用的级别。

表 29. Java 跟踪级别选项			
配置的跟踪级别	Java 日志记录跟踪级别	描述	
关	OFF	不执行任何日志记录。	
错误	SEVERE	跟踪 Java 应用程序中发生的问题。	
<b></b> 鬱日	WARNING	跟踪错误和潜在错误。	

表 29. Java 跟踪级别选项 (续)			
配置的跟踪级别	Java 日志记录跟踪级别	描述	
参考	INFORMATION	跟踪关于 Java 应用程序的重要信息。	
最低级别调试	FINE	跟踪分析 Java 应用程序行为所必 需的高级别详细信息。	
中等级别调试	FINER	跟踪关于 Java 应用程序的程序流的详细信息。	
最高级别调试	FINEST	跟踪关于 Java 应用程序的所有详细信息。	
全部	ALL	跟踪所有消息。	

在此示例中, Java 应用程序创建的日志文件的名称为 k91\_trace0.log。对于多实例代理程序, 实例名称 将包括在日志文件名中。

**注:** 请不要将消息写至标准错误或标准输出。在 Windows 系统上,这些消息将丢失。在 UNIX 和 Linux 系统上,此数据将写入一个不回绕的文件。

#### 收集采样属性组数据

采样属性组(收集一个或多个数据行的属性组)的类包含 collectData 方法,例如 Sampled\_Data.collectData。每当代理程序请求获取数据时,都将调用此方法。

属性组的帮助程序类定义了一个名为 Attributes 的内部类。对于该属性组中定义的每个属性,这个类都有一个对应的字段。由于派生属性由代理程序进行计算,因此未包括在内。属性字段的数据类型是 Tivoli Monitoring 属性类型的 Java 对等项,如<u>第 144 页的表 30</u> 所示。

表 30. 属性字段的数据类型及其 IBM Tivoli Monitoring 属性类型对等项			
Tivoli Monitoring 类型	属性字段的数据类型		
字符串	String		
32 位数字(不进行小数调整)	int		
64 位数字(不进行小数调整)	long		
数字(非零小数调整)	double		
时间戳记	Calendar		

collectData 方法必须完成下列任务:

- 1. 从受监视的资源收集相应的数据。
- 2. 创建 Attributes 对象。
- 3. 将数据添加到该 Attributes 对象的字段。
- 4. 调用 Attributes.setAttributeValues 方法,以便将数据复制到内部缓冲区。
- 5. 根据需要,对每个数据行重复第1步到第4步。(可以同时跳过第1步到第4步,而不返回任何行。在这种情况下,Performance Object Status 表的"错误码"列的值为 NO\_INSTANCES\_RETURNED。有关错误码的更多信息,请参阅<u>第147页的『错误码』</u>。
- 6. 调用 AgentConnection.sendData 以便将数据发送到代理程序,或者调用 sendError 以废弃通过调用 setAttributeValues 复制的数据并改为发送错误码。

您必须从资源收集数据(第1步),从而替换在生成的应用程序中使用的样本数据。

要填充该 Attributes 对象,您可以使用 Attributes 构造函数来传递数据(正如生成的应用程序中的做法)。 另外,也可以使用不带自变量的构造函数来创建 Attributes 对象,然后将收集到的属性值赋予该 Attributes 对象的字段。字段与属性同名,但字段名称以小写字母开头。

#### 收集子节点的采样数据

如果采样属性组位于子节点中,那么您可能正在监视多个资源(对于每个子节点,都存在不同的资源)。您必须确定要从哪些资源收集数据。必须有一个或多个用于标识所监视资源的配置属性。

对于本示例,假定配置属性 K91\_INSTANCE\_KEY 中包含的值指示必须从中收集数据的资源。

请通过下列步骤来查找正确的资源:

- 1. 通过调用 AgentConnection.getConfiguredSubnodeInstanceIDs,获取所有已配置的子节点的 实例标识。每个已配置的子节点都具有唯一的实例标识。
- 2. 对于每个实例标识, 通过调用 AgentConnection.getSubnodeConfigurationProperty 获取配置 属性 K91\_INSTANCE\_KEY。
- 3. 找到 K91\_INSTANCE\_KEY 中的值所表示的资源。

这些步骤可以在<u>第144页的『收集采样属性组数据』</u>中详细描述的步骤序列之前,在 collectData 方法中 完成。

另外,您也可能想在属性组类构造函数中执行这些步骤,并建立从实例标识到资源的直接映射。 Sampled\_Subnode构造函数是属性组类构造函数的一个示例。此过程还使您有机会创建句柄或打开可以在 代理程序的整个生命周期内使用的连接。创建句柄或打开连接可以更有效地访问资源。

生成的代码将在构造函数中创建类型为 MonitoredEntity 的样本资源对象,并将这些对象添加至 configurationLookup 映射。您必须移除 MonitoredEntity 内部类,并将 MonitoredEntity 对象 替换为用于访问您自己的资源的对象。如果您选择在 collectData 方法中执行整个查找过程,那么可以从 该类中移除 configurationLookup 映射。

如果您选择使用构造函数,从而将子节点实例标识映射到资源,那么 collectData 方法中的步骤如下所示:

1. 通过调用 Request.getSubnodeInstanceID,从请求参数中检索子节点的实例标识。

2. 从构造函数中创建的映射中检索资源对象。

3. 执行第 144 页的『收集采样属性组数据』中详细描述的步骤序列,以便将数据发送到代理程序。

在 Agent Builder 示例中选择了任意子节点属性(在本例中,这是 K91\_INSTANCE\_KEY)。如果选择了错误的属性,或者需要多个属性才能标识正确的资源,那么您必须选择这些属性以标识该资源。

#### 发送事件

对于生成事件的属性组而言,不会定期调用 collectData 方法。在资源发布事件时,您的应用程序将发送这些事件。

下面是生成事件的一个示例:针对基于事件的属性组生成的代码创建并启动一个线程,该线程从名为 SampleEventClass 的内部类中运行。此示例中使用的基于事件的属性组为 Event\_Data 类。此线程将定 期唤醒并发送事件。如果您希望定期轮询资源以查找事件,那么可以使用 Event\_Data 类在生成时具有的 结构:

- 1. 在 Event\_Data 构造函数中, 创建并启动一个线程。
- 2. 在此线程的 run 方法中,进行循环,直到代理程序终止为止。
- 3. 休眠一段时间, 然后查找事件。您可能想将轮询时间间隔 5000 毫秒更改为对代理程序有意义的值。
- 4. 确定是否发生了一个或多个事件。生成的应用程序不会检查是否发生了事件,而会始终发布单一事件。
- 5. 对于每个必须发布的事件,获取要发布的事件数据。
- 6. 创建并填充 Attributes 对象(与 collectData 方法对采样属性组进行的处理相似)。
- 7. 调用 Attributes.sendEventData 方法。事件由单一的行组成,因此,每次只能发送单一事件。

另外,如果您正在使用通过自己的线程来报告事件的 Java API,那么可以在 Event\_Data 构造函数中初始 化该线程。您还可以向资源的事件处理机制注册您自己的事件处理对象。在事件处理程序中,使用以下步骤:

- 1. 获取要发布的事件数据。
- 2. 创建并填充 Attributes 对象。
- 3. 调用 Attributes.sendEventData 方法。

在这种情况下,不必在 Event\_Data 类中创建自己的线程,也不需要 SampleEventClass 类。

#### 在子节点中发送事件

检测到子节点属性组的事件时, Java 应用程序必须将该事件发布到正确的子节点。

在本示例中,假定配置属性 K91\_INSTANCE\_KEY 中包含的值指示可以生成事件的资源实例。并且,假定 K91\_INSTANCE\_KEY 属性的值与要在事件中发布的数据一起检索。要检索属性和数据,Java 应用程序应执 行以下步骤:

- 1. 获取要发布的事件数据以及"实例键"。
- 2. 创建并填充 Attributes 对象。
- 3. 通过调用 AgentConnection.getConfiguredSubnodeInstanceIDs, 获取所有已配置的子节点实 例标识的列表。
- 对于每个子节点实例,通过调用 AgentConnection.getSubnodeConfigurationProperty 访存 K91\_INSTANCE\_KEY 的值。
- 5. 发现 K91\_INSTANCE\_KEY 的值与随事件数据获取的值匹配时,记录相应的子节点实例标识。
- 6. 在传递所记录的子节点实例标识的情况下,调用 Attributes.sendSubnodeEventData。

生成的应用程序不会执行第4步和第5步中描述的查找操作, 而是将事件发布到每个子节点的属性组。对于 生产代理程序而言, 这可能不是正确的行为。

#### 执行操作命令

您可以在 Tivoli Enterprise Portal 中定义执行操作命令,也可以使用 tacmd createaction 命令来定义这些操作。并且,可以将这些操作导入到代理程序的 Agent Builder 项目中,以便在安装代理程序时创建这些操作。有关导入执行操作命令的更多信息,请参阅<u>第 217 页的『第 15 章 导入应用程序支持文件』</u>。

生成的 Java 应用程序将针对任何以代理程序产品代码开头的操作(例如 K91Refresh)进行注册。此注册 在主帮助程序类 (MainClassBase)的 registerActionPrefix 方法中完成。如果要注册其他前缀或者 完全不针对任何操作进行注册,请覆盖 (MainClassBase)中的 registerActionPrefix。

当代理程序要运行以代理程序所注册的前缀开头的操作时,将调用 MainClass.takeAction 方法。您可以添加代码以调用 Request.getAction()并执行适当的操作,然后调用

AgentConnection.sendActionReturnCode 以发送操作返回码。返回码为 0 表示操作成功,任何其他 返回码均表示操作失败。

#### 处理异常

collectData 和 takeAction 方法可以抛出任何 Java 异常,因此,您可以允许收集代码抛出异常而不捕获这些异常。当帮助程序类接收到异常时,将调用 handleException 方法(对于 collectData)或 handleActionException 方法(对于 takeAction)。

对于 collectData 异常,您必须在异常发生时或者在收集数据期间出现任何问题时调用 AgentConnection.sendError。生成的应用程序传递错误码 GENERAL\_ERROR。但是,您必须将此错误 码替换为由代理程序定义且最能说明所遇到问题的错误码。有关添加错误码的更多信息,请参阅<u>第 140 页的</u> <u>『13』</u>。

对于 takeAction 异常, 您必须调用 AgentConnection.sendActionReturnCode, 且使用非零返回 码。

某些 AgentConnection 方法会抛出派生自

com.ibm.tivoli.monitoring.agentFactory.customProvider.CpciException的异常。如果 在收集数据期间抛出了CpciException,那么不会调用 handleException 方法,这是因为帮助程序类 将处理该异常。

**注:** 如果您选择在 collectData 方法内捕获异常,而不使用 handleException 方法,请确保重新抛出任 何 CpciException。确保重新抛出 CpciException 以使其可以由基类进行处理。

#### 错误码

对异常或其他资源错误的典型响应是,通过调用 AgentConnection.sendError 方法将错误码发送到代 理程序。基于事件的属性组的错误随时都可以发送。采样属性组的错误只能作为对收集数据请求的响应进行 发送,并且替代 sendData 调用。

如果将错误发送到代理程序,那么将发生下列情况:

- 1. 错误消息将记录在代理程序跟踪日志中。此错误消息包含错误码以及针对该错误码定义的消息。
- 2. 存在 Performance Object Status 查询,您可以查看此查询以获取属性组的状态信息。"错误码"列将设置为针对所发送错误定义的错误码类型。在代理程序成功接收到此属性组的数据后,错误状态将清除。如果您使用 sendData 调用来应答收集数据请求,但未包括任何数据行,那么"错误码"列将显示NO\_INSTANCES\_RETURNED。

下表描述了一些在某些情况下有可能出现的代理程序内部错误码:

表 31. 代理程序的内部错误码	
错误码	描述
NO_ERROR	此属性组目前没有任何问题。
NO_INSTANCES_RETURNED	Java 应用程序已响应数据收集请求,但未提供任何 数据。未提供数据不是错误。通常,这表示此属性组 未监视任何资源实例。
OBJECT_NOT_FOUND	代理程序试图收集未通过客户机 API 注册的属性组的数据。此错误可能表示应用程序未能启动,或者应用程序在代理程序尝试收集数据时未启动属性组注册。
OBJECT_CURRENTLY_UNAVAILABLE	应用程序向代理程序发送了未在全局错误码列表中定 义的错误码。
GENERAL_ERROR	从应用程序收集数据时发生问题,这通常是因为应用 程序未在超时时间间隔内对请求作出应答。有关更多 详细信息,请参阅代理程序跟踪日志。
	另外,应用程序也可以指定 GENERAL_ERROR 作为错误码,但最好定义更详细的错误码。

#### 对代理程序所作的更改

对代理程序所作的某些更改要求对 Java 应用程序进行相应更改。如果结构更改非常复杂,那么可以先删除 任何或全部 Java 源文件,然后再保存代理程序。如果您想在未进行定制的情况下重新开始,那么也可以删 除这些文件。

下表描述了在 Agent Builder 中进行某些更改之后,在保存该代理程序时需要对 Java 应用程序源文件进行的 修改。

表 32. 要求对 Java 源代码进行修改的代理程序更改				
代理程序更改	Agent Builder 执行的操作	需要对 Java 源代码进行的手动更改		
更改主类的包名	·在新的包结构中生成所有的类。 ·从旧包中移除所有的帮助程序 类。	<ul> <li>·将主类及属性组类的内容从旧包中的类移植到新包中的类。</li> <li>·在迁移完成后,从旧包中移除这些类。</li> </ul>		
更改主类的名称	·创建新的主类。 ·移除旧的主帮助程序类。	·将主类的内容移植至新类。 ·更新属性组类中对类名的引用。		
添加 Java API 属性组	<ul> <li>·针对新的属性组创建类。</li> <li>·在主帮助程序类中,针对新的属性组添加注册。</li> </ul>	使用属性组类中的定制逻辑覆盖样本代码。		
移除 Java API 属性组	从主帮助程序类中移除注册。	·移除属性组类,或者将定制逻辑 移植到另外某个类。 ·移除属性组帮助程序类。		
重命名 Java API 属性组	<ul> <li>·针对属性组的新名称创建类。</li> <li>·在主帮助程序类中,更新已重命 名的属性组的注册。</li> </ul>	<ul> <li>·将具有旧名称的属性组类中的定制逻辑移植至具有新名称的属性组类。</li> <li>·移除具有旧名称的属性组类。</li> <li>·移除具有旧名称的属性组类。</li> <li>·移除具有旧名称的属性组帮助程序类。</li> </ul>		
向 Java API 属性组添加属性	在属性组帮助程序类中更新 Attributes 内部类。	在属性组类中收集新属性的数据。		
从 Java API 属性组中移除属性	在属性组帮助程序类中更新 Attributes 类。	在属性组类中移除先前属性的数据 收集。		
重命名 Java API 属性组中的属性	在属性组帮助程序类中更新 Attributes 类中的属性名称。	更新 Attributes 类中的所有对该属 性名称的引用(由于使用了具有位 置参数的 Attributes 构造函数,因 此通常不存在任何引用)。		
对 Java API 属性组中的属性进行 重新排序	在属性组帮助程序类中更新 Attributes 类中的属性顺序。	更新全部 Attributes 构造函数调用 中的自变量顺序。		

如果使用"Eclipse 重构 - 重命名"操作,那么上表提及的某些更改可以简化。请先对所有受影响的名称(包括帮助程序类名)使用此操作,然后再保存所更改的代理程序。

#### Java API 的使用

在生成的 Java 应用程序中,将始终使用 Java API 与代理程序进行通信。通常,您与 Java API 进行的唯一直接交互是修改现有方法调用的参数。例如,将发布的错误码由 GENERAL\_ERROR 更改为代理程序中定义的错误码。

如果要使用 Java API 进行更广泛的编码,您可以从 Eclipse 文本编辑器中查看 Javadoc。在编辑 Java 代码时,可以通过执行下列步骤查看 Javadoc:

- 1. 突出显示 API 中的包名、类名或方法名。
- 2. 按下 F1 以打开 Eclipse 帮助视图。
- 3. 选择 Javadoc 链接。

您也可以通过将鼠标悬停在类名或方法名上查看 Javadoc 中的简要描述。此 API 的 Javadoc 也可以在 Tivoli Monitoring Knowledge Center 上找到,请参阅 Javadoc。

Java API 的类包含在 cpci.jar 中。在创建包含 Java API 属性组的代理程序时, cpci.jar 文件将自动添加到项目的 Java 构建路径。在导入包含 Java API 属性组的代理程序时,也将添加该文件。将 Java API 属性组称加到现有代理程序时,也将添加该文件。另外, cpci.jar 还将自动地与每个包含 Java API 属性组的代理程序打包在一起,并添加到 Java 应用程序的 CLASSPATH。

## Java API 配置

在代理程序中定义 Java API 数据源时,将为您创建部分配置属性。

如果在代理程序中定义了 Java API 数据源,那么该代理程序必须使用 Java 来连接到 Java API 服务器。 Java 配置属性将自动添加到该代理程序中。下列 Java 配置属性特定于代理程序运行时配置:

表 33. Java 配置属性					
姓名	有效值	是否必需	描述		
Java 主目录	目录的标准路径	否	指向 Java 安装目录的标 准路径。		
Java 跟踪级别	选项	是	请使用此属性来指定 Java 提供程序所使用的 跟踪级别。		
JVM 自变量	字符串	否	请使用此属性来指定 Java 虚拟机的可选自变 量列表。		
外部 JAR 文件的类路径	字符串	否	包含所有未随代理程序一 起提供但却是运行时客户 机操作所必需的 JAR 文 件的路径。		

这些配置变量位于 Agent Editor 中 "运行时配置信息"页面上的 Java 虚拟机 (JVM) 配置和 Java API 配置 之下。

#### 测试 Java 应用程序属性组

您可以对在 Agent Builder 中创建的 Java 应用程序属性组进行测试。

#### 开始之前

**限制:** 与大部分其他属性组不同,您在创建 Java 应用程序属性组时,无法对其进行测试。您可以先将此属性 组添加到代理程序并保存代理程序,然后再测试此属性组。保存代理程序将致使生成此属性组的 Java 代 码。

#### 过程

1. 创建代理程序之后,请在 Agent Editor 的"数据源定义"页面上选择属性组,然后单击测试。

有关 Agent Editor 的更多信息,请参阅 第 17 页的『第 4 章 使用 Agent Editor 修改代理程序』

在上述两个步骤中的某个步骤中单击测试之后,将显示"测试 Java 客户机"窗口。

- 2. 可选: 在开始测试之前,请设置环境变量、配置属性及 Java 信息。有关更多信息,请参阅<u>第 191 页的</u> <u>『属性组测试』</u>。有关缺省 Java 运行时配置属性的更多信息,请参阅 <u>第 149 页的『Java API 配置』</u>。
- 3. 单击启动代理程序。此时将显示一个窗口,表明代理程序正在启动。
- 4. 要模拟来自 Tivoli Enterprise Portal 或 SOAP 的代理程序数据请求,请单击**收集数据**。

代理程序会监视 Java 客户机的数据。"测试 Java 客户机"窗口将显示所返回的全部数据。

5. 可选: 如果返回的数据未如预期,请单击检查结果。

"数据收集状态"窗口将会打开,并显示有关该数据的更多信息。<u>第 239 页的『Performance Object</u> Status 节点』对 Data collection Status 窗口所收集并显示的数据作了描述。

6. 通过单击停止代理程序停止代理程序。

7. 单击确定或取消以退出"测试 Java 客户机"窗口。单击确定将保存您所作的全部更改。

#### 相关概念

第191页的『在 Agent Builder 中测试代理程序』

使用 Agent Builder 创建代理程序之后,可以在 Agent Builder 中测试该代理程序。

# 第7章根据现有来源创建数据集

至少存在一个数据集时,您可使用现有数据集中的数据来创建新数据集。

"**代理程序初始数据源**"页面和"数据源位置"页面都提供了创建新数据集的选项。您可以通过下列方法使 用现有数据源来创建数据集:

1. 连接来自两个现有数据集(属性组)的数据。有关更多信息,请参阅<u>第151页的『连接两个属性组』</u>。
 2. 过滤来自某个现有数据集(属性组)的数据。有关更多信息,请参阅<u>第155页的『创建过滤属性组』</u>。
 提示:只有在创建两个或两个以上数据集之后,连接两个数据集的选项才可用。

## 连接两个属性组

从两个其他属性组创建一个属性组。

#### 关于此任务

代理程序从两种不同类型的数据源收集数据时,连接式属性组非常有用。例如,代理程序可以收集数据 WMI 和 PerfMon,或 SNMP 和脚本数据源。各组属性在一个 Tivoli Enterprise Portal 视图中搭配使用时可能更加 有用。

例如,假设您的属性组定义如下:

First\_Attribute\_Group index integer trafficRate integer errorCount integer

Second\_Attribute\_Group index2 integer name string traffic string

一个定义向您提供了计数器(例如: Perfmon),另一个向您提供了识别信息。没有属性组本身对您有用。 但是,如果您可以使用索引将这两个属性组组合起来以与每个属性组的相应行进行匹配,那么会得到一个更 有用的属性组。您可以使用组合的属性组来同时显示名称、类型和度量。

此机制还能用于将标记添加到通过普通属性组收集的信息内。然后在检测到问题时能够在事件系统中轻松地与信息进行关联。例如,某公司想通过收集公共数据,并使用公共情境来监视服务器的运行状况,从而管理其所有服务器。此公司还希望能通过更多的信息(指示正在特定服务器上运行哪些应用程序)来识别服务器。此公司想要控制每台服务器上使用的值,但是不想为各个应用程序创建不同的代理程序。它可以在其单个代理程序中创建一个附加属性组来实现此控制,如下所示:

Application_Information	
application_type	integer
application_name	string
application_group	string

此属性组可定义为脚本属性组,用于从代理程序配置收集值。您可以为每个代理程序实例指定不同的值,并 使用一个代理程序来管理他们的所有系统。此属性组随即将连接至所有可能需要该应用程序信息的源属性 组。然后该信息将在 Tivoli Enterprise Portal、情境、事件和已仓储数据中可用。

连接两个属性组时,将创建第三个属性组。此属性组包含源属性组中包含的所有属性。

连接操作的结果会因各个源属性组支持的行数不同而不同。如果两个属性组都定义为只返回一行数据,那么 生成的连接式属性组将有一行数据。单个行包含来自这两个源属性组的所有属性。

表 34. 源属性组 1(单行)				
属性1	属性 2	属性3		
16	部分文本	35		

表 35. 源属性组 2(单行)			
属性 4	属性 5	属性6	属性 7
5001	更多数据	56	35

表 36. 生成的连接						
属性1	属性 2	属性 3	属性 4	属性 5	属性 6	属性 7
16	部分文本	35	5001	更多数据	56	35

假设一个源属性组定义为只返回一行(单行),而另一个源属性组可返回多行(多行)。生成的连接式属性 组包含与多行源属性组相同的行数。来自单行属性组的数据将添加到多行属性组中的各行。

表 37. 源属性组 1(单行)				
属性1	属性 2	属性3		
16	部分文本	35		

表 38. 源属性组 2(多行)				
属性 4	属性 5	属性6	属性 7	
user1	path1	56	35	
user2	path2	27	54	
user3	path3	44	32	

表 39. 生成的连接						
属性1	属性 2	属性 3	属性 4	属性 5	属性 6	属性 7
16	部分文本	35	user1	path1	56	35
16	部分文本	35	user2	path2	27	54
16	部分文本	35	user3	path3	44	32

最后,假设两个源属性组都定义为返回多行。您必须识别要执行连接操作的每个源属性组的属性。生成的属性组将包含多个数据行,其中,第一个属性组的属性值与第二个属性组的属性值相匹配。

表 40. 源属性组 1(多行)			
属性1	属性 2	属性3	
16	部分文本	35	
27	更多文本	54	
39	其他字符串	66	

表 41. 源属性组 2(多行)				
属性4	属性 5	属性 6	属性 7	
user1	path1	56	35	
user2	path2	27	54	
user3	path3	44	32	

表 42. 生成的连接(按属性 3 和属性 7 连接)						
属性1	属性 2	属性 3	属性 4	属性 5	属性 6	属性 7
16	部分文本	35	user1	path1	56	35
27	更多文本	54	user2	path2	27	54

如果代理程序中定义了任何可用性过滤器,还可以使用 Agent Builder 将用户定义的属性组连接到"可用性"属性组。有关"可用性"属性组中包含的数据的更多信息,请参阅<u>第 235 页的『可用性节点』</u>。通过单击右键访问数据源树的菜单并选择**连接式属性组**,您可以创建此类型的属性组。

过程

在"数据源定义"页面上,右键单击其中一个要连接的属性组,然后选择连接式属性组。
 仅当至少定义了两个属性组时才显示该选项。定义了可用性过滤器即认为是定义了属性组。
 "属性组信息"页面随即显示。

Attribute Group Information	
Attribute Group Information Enter the identification information for this attribute group.	
Attribute group name	
Help text	
- Join Information	Attribute Group Two
Attribute_Group_1	✓
<ul> <li>Produces a single data row</li> <li>Can produce more than one data row</li> <li>Produces events</li> </ul>	<ul> <li>Produces a single data row</li> <li>Can produce more than one data row</li> <li>Produces events</li> </ul>
Attribute to join on	Attribute to join on
0	OK Cancel

图 18. "属性组信息"页面"属性组信息"窗口

2. 在"**连接信息**"区域中,选择要连接的两个属性组。通过从**属性组一**和**属性组**二列表中的群组中进行选择,选择属性组。

对于每个属性组,系统将为您选择**生成单一数据行**或**可生成多个数据行**。此选择已锁定并取决于最初如 何定义这些源属性组。

注:对于可以连接式属性组有以下限制:

- ·不能将一个子节点类型中的属性组连接到另一个子节点类型中的属性组。
- ·只能将事件属性组连接到单行非事件属性组。
- a) 当两个属性组在**连接所依据的属性**下都显示了**可生成多个数据行**时,请为每个属性组选择要用作连接 依据的属性。

属性组名称和帮助字段将使用所选属性组的信息进行填写。您可以根据需要来更改这些条目。

3. 单击确定。

#### 结果

您创建的连接式属性组将添加到"数据源定义"页面的"属性组信息"区域中。

## 处理连接式属性组中的属性

使用连接式属性组中的属性可以对如何处理这些属性实施规则。

#### 删除属性组

如果某个属性组在连接式属性组中被引用,那么除非连接式属性组也要被删除,否则不能删除该属性组。

#### 删除属性

如果某个属性的父属性组在连接式属性组中被引用,并且符合以下某项,则无法删除该属性:

·该属性被定义为连接式属性组中的连接式属性。

·该属性在连接式属性组中的任何派生属性中使用。

无法删除连接式属性。只能从连接式属性组删除派生属性(如果添加了这些派生属性)。

#### 重新排序属性

连接式属性的顺序固定为源属性的顺序。无法对连接式属性列表重新排序。只能对派生属性(如果有)重新 排序。

落实代理程序的版本后,源属性和派生属性就无法重新排序,并且不可移除。添加在新版本的代理程序中的 属性(源或派生属性)将跟在所有落实的属性之后。有关更多信息,请参阅<u>第 29 页的『落实代理程序的版本』</u>。

#### 添加属性

无法显式添加新的连接式属性。只能显式创建派生属性。

#### 移除可用性过滤器

如果在连接式属性组中引用了"可用性"属性组,那么不能移除最后一个可用性过滤器。

## 连接式属性

处理与连接式属性有关的信息

#### 过程

·您可以更改连接式属性的属性名称和帮助文本,以使其与源属性不同: a) 在**数据源定义**页面的**属性组信息**窗格中,选择连接式属性组中的属性。 b) 输入新名称和帮助文本。

- 通过选中或取消选中在 Tivoli Enterprise Portal 中显示属性复选框,可以使连接式属性显示或不显示在 Tivoli Enterprise Portal 中。此复选框位于数据源定义页面的连接式属性信息部分中。此选项与源属性是 否显示在 Tivoli<sup>®</sup> Enterprise Portal 中无关。
- · 通过选中**键属性**复选框,可以将 Tivoli Enterprise Portal 中显示的任何属性或属性组合标记为键属性。此 选项与属性是否是源属性组中的键属性无关。此选项与源属性是否显示在 Tivoli<sup>®</sup> Enterprise Portal 中也 无关。
- · 连接式属性的属性类型信息是从源属性取得并且无法在连接式属性中更改。在 Agent Editor 的 连接式属 性组信息部分 (<u>第 155 页的图 19</u>)中,单击**找到源属性**即可转到源属性。

Attribute name At	ttribute_B					
Help At	ttribute_B					
✓ Display attribute Key attribute	e in the Tivoli	Enterprise Portal				
Join Attribute Inf	ormation —					
Source attribute Source attribute	group: AG3 : Attribute_B	i.			Locate sou	rce attribute
Attribute type —						
	Size	<ul> <li>32 bits</li> </ul>		🔿 64 bits		
<ul> <li>String</li> <li>Numeric</li> </ul>	Purpose	<ul> <li>Gauge</li> <li>Delta</li> </ul>	<ul> <li>Counter</li> <li>Percent change</li> </ul>		<ul> <li>Property</li> <li>Rate of change</li> </ul>	
◯ Time stamp	Scale	Decimal adjustment 0				
	Range	Minimum None		Maximum No	ne	
Enumerations —			· ·			, ,

#### 图 19. 找出源属性信息

对源属性组所作的更改会反映到连接式属性中。如果源属性组发生更改,那么连接式属性组的那些属性 将自动更新。如果将另一个属性组设置为源属性组,那么也会发生此自动更新。对源属性类型所做的更 改将复制到连接式属性。对源属性名称或帮助文本所作的更改将复制到连接式属性。但是,在更改连接 式属性的名称或帮助文本后,不会复制这些源属性更改。

## 创建过滤属性组

通过从现有属性组中过滤数据行来创建过滤属性组(数据集)。如果现有数据集返回多行,可以创建返回单行的已过滤组以用于 IBM Cloud Application Performance Management。

#### 关于此任务

过滤属性组包含与源属性组相同的列,但排除了其中一些行。它使用选择公式来确定要包含的行。

要为 Cloud APM 提供状态和摘要信息,需要使用返回单行的数据集。有关详细信息,请参阅<u>第 187 页的</u> <u>『第 12 章 为 Cloud APM 准备代理程序』</u>。如果源信息位于返回多行的数据集中,可以创建返回单行的过滤 属性组。

例如,进程、Windows 服务和命令返回码数据源在单个"可用性"数据集中以行的形式提供信息。您可以在选择公式中使用"名称"字段来创建过滤属性组。该组包含必要应用程序的状态。将其定义为返回单行。然后,可以使用该属性组作为 Cloud APM 的摘要数据集。

如果您希望将基本数据源查询返回的数据分到不同的组中,那么过滤属性组也很有用。Windows性能监视器、SNMP和WMI是此类数据源的示例。

例如,假设数据源可返回下列数据:

Name	Туре	Size	Used	Free
Memory	MEM	8	4	4
Disk1	DISK	300	200	100
Disk2	DISK	500	100	400

这是报告系统上现有存储器(包括内存和磁盘空间)的表。您可能希望将该表拆分成两个独立的表,分别对 应内存和磁盘。您可以创建两个基本属性组来拆分该表。每个基本属性组收集相同的数据并过滤掉不需要的 行。但是,这并不是效率最高的做法。您可以改为定义一个同时返回内存和磁盘使用情况数据的基本属性 组。然后,定义两个过滤属性组。这两个组都使用同一个基本表作为源。其中一个包含 Type=="MEM" 的过 滤器,另一个包含 Type=="DISK" 的过滤器。

在示例中,对于 Type=="MEM" 的过滤属性组,返回的数据是:

Name Type Size Used Free Memory MEM 8 4 4

对于 Type=="DISK",返回的数据是:

Name	Type	Size	Used	Free
Disk1	DÍSK	300	200	100
Disk2	DISK	500	100	400

注:数据基于事件的属性组无法用来创建过滤型属性组。只能使用包含样本数据的属性组。

#### 过程

1. 在"代理程序初始数据源"页面或"数据源位置"页面上的监视数据类别区域中,单击现有数据源。

注:

·您可以使用"新建代理程序"向导转到"代理程序初始数据源"页面。有关更多信息,请参阅<u>第13页</u> <u>的『第3章 创建代理程序』</u>。

·您可以通过在 Agent Editor 的"数据源定义"页面中右键单击代理程序并选择添加数据源,转到"数据源位置"页面。

- 2. 在数据源区域中,选择过滤属性组的数据行。
- 3. 单击下一步。

"过滤器信息"页面随即显示。

- 4. 从列表中选择源属性组。
- 5. 输入选择公式,以便对来自所选属性组的数据进行过滤。 例如,在前面显示的"过滤器信息"页面中,选择公式根据 Type 属性等于 "DISK" 对数据行进行过滤。 Type 属性与 "DISK" 不匹配的数据行将被废弃。您输入的选择公式的求值结果必须是布尔值 true 或 false。

**注:** 在"**过滤器信息**"页面中,您可以单击**编辑**,以使用公式编辑器来输入或修改公式。有关公式编辑器 的更多信息,请参阅<u>第 39 页的『Formula Editor』</u>。

- 6. 单击**下一步**。
- 7. 选择生成单一数据行或可生成多个数据行。

a) 如果选择了可生成多个数据行,请从列表中选择一个或多个键属性。

8. 单击完成。

# 第8章创建导航器组

在 IBM Tivoli Monitoring 环境中,使用导航器组将若干相关数据源(属性组)分组到一起,以便能够创建工 作空间来显示合并了数据源的视图。在基本代理程序级别使用"新建代理程序"向导创建代理程序时,可以 创建导航器组。使用"新建代理程序组件"向导定义子节点时也可以创建导航器组。

#### 关于此任务

例如,您也许能够从多个数据源收集文件系统数据。创建一个显示不同数据源的所有文件系统数据视图的工作空间可能十分有用。

要在 Tivoli Enterprise Portal 中隐藏数据源,导航器组也是一种很好的方法。您可能决定如果将两个数据源 合并以创建一个新的组合数据源,那么从这两个数据源收集的度量值最有用。您可能只想在合并数据源中看 到组合的数据。您可以创建包含全部三个数据源的导航器组,并创建一个工作空间,其中包含只显示合并数 据源的视图。可以有效地在 Tivoli Enterprise Portal 中隐藏两个原始数据源。请参阅<u>第 151 页的『第 7 章 根</u> 据现有来源创建数据集』以了解有关合并数据源的信息。

**注:** 将数据源分组到导航器组时, Tivoli Monitoring 不会使查询与该导航器组相关联。而是假设您为导航器组 定义了缺省工作空间, 从而以有用的格式显示数据源。

导航器组可以在基本代理程序或子节点中定义。导航器组不得包含其他导航器组。

导航器组在 IBM Cloud Application Performance Management 环境中不起作用。

#### 过程

1. 执行下列其中一个步骤:

- · 使用"代理程序"向导创建新代理程序时,在"代理程序初始数据源"页面上,请单击监视数据类别 区域中的数据源分组。
- · 对于现有代理程序,请在 Agent Editor 中执行以下步骤:
  - a. 单击数据源选项卡,以打开"数据源定义"页面。
  - b. 选择代理程序并单击添加至选定项。
  - c. 在"数据源位置"页面上的监视数据类别区域中,单击数据源分组。
- 2. 在数据源区域中,单击导航器组。
- 3. 单击下一步。
- 4. 在"导航器组信息"页面上,输入导航器组名称以及要与此名称关联的帮助文本,然后单击下一步。
  注: 在某些情况下, Agent Builder 将自动创建导航器组。以下导航器组名称是保留名称:
  可用性
- 5. 在"第一个导航器组数据源"页面上,选择新导航器的第一个监视数据源。在监视数据类别列表中单击 某个类别,并在数据源列表中单击某个数据源。然后单击下一步。

**提示:**您可以照常创建数据源。或者,单击**现有数据源**并选择将已创建的一个或多个数据源移入导航器组。

- 6. 如果要在导航器组中创建数据源,请在"数据源定义"页面上选择该导航器组,然后单击添加至选定 项。
- 7. 如果要将现有数据源移入导航器组,请在"数据源定义"页面上选择该导航器组,单击添加至选定项, 然后在"导航器组数据源"页面上选择现有数据源。在"当前定义的数据源"页面中,选择数据源。
- 8. 如果要从导航器组移除数据源,请在"数据源定义"页面上执行下列其中一个步骤:
  - ·选择数据源,然后将其拖至数据源树的根。
  - ·选择数据源,然后单击移除。
- 9. 如果要创建导航器组,请在"数据源定义"页面上执行下列其中一个步骤:

- · 单击添加至代理程序。
- ·选择子节点,然后单击**添加至选定项**。

# 第9章使用子节点

您可以使用子节点从单个代理程序实例监视多个应用程序组件。

通过使用子节点,您可以构建单个代理程序来完成以下任务:

- ·监视系统上运行的每个软件服务器实例,而无需针对每个软件服务器实例使用一个单独的代理程序实例。
- ·监视若干个不同的远程系统,而无需针对每个远程系统使用一个单独的代理程序实例。
- ·从一个代理程序监视多种不同类型的资源,而无需构建并部署若干个不同的代理程序。
- ·在 IBM Tivoli Monitoring 中的 Tivoli Enterprise Portal 物理导航树中显示额外的级别,以允许进一步分组 和定制。而且,您可以定义受管系统组,按情境进行进一步细分。
- · 在 IBM Cloud Application Performance Management 中,提供若干不同的资源,显示不同的摘要和详细 信息仪表板。子节点资源可以显示为代理程序资源的对等项或子组件。您可以将这些资源单独包括在应用 程序中。

您可以在 Agent Builder 中创建子节点类型。每种类型必须对应于代理程序所能够监视的不同资源类型。请 针对特定的受监视资源,将数据源和数据集添加到子节点类型。

将代理程序部署到受监视主机并进行配置时,可以为每种子节点类型创建一个或多个实例。子节点的每个实例都必须对应于服务器实例、远程系统实例或指定让该子节点类型监视的任何资源实例。单一子节点类型的所有子节点实例都有相同形式的属性组和工作空间。但是,每个子节点实例都有来自特定受监视资源的数据。

在受监视主机上配置代理程序时,您可确定子节点实例的数量。某些配置数据可作为整体应用于代理程序, 而其他配置数据应用于单个子节点实例。请对每个子节点实例采用与其他子节点实例不同的配置方式,确保 它们不会监视完全相同的资源,也不会显示完全相同的数据。

在 IBM Tivoli Monitoring 环境中,子节点实例显示在 Tivoli Enterprise Portal 的导航器树中的代理程序下。 工作空间显示子节点实例所生成的数据,情境可以分发到子节点的一个或多个实例。将会自动创建包含子节 点所有实例的受管系统列表,就像为代理程序创建的"受管系统列表"一样。

在 IBM Cloud Application Performance Management 环境中,代理程序实例和子节点实例都可以显示为受监视资源。每个子节点实例都会成为单独的资源。有关详细信息,请参阅<u>第 164 页的『IBM Cloud</u> Application Performance Management 中的子节点』。

由于使用 Agent Builder 构建的代理程序会根据配置值来创建子节点实例,因此这些子节点的寿命与代理程序相同。仍然只有一个针对代理程序的脉动信号,而不存在针对每个子节点的单独脉动信号。因此,通过使用子节点,可以显著加大监视环境的潜在规模。替代方法是使用多个代理程序实例,但是这样会限制 IBM Tivoli Monitoring 或 IBM Cloud Application Performance Management 环境的潜在规模。

添加或移除子节点要求重新配置代理程序。要重新配置代理程序,您需要将其停止并重新启动,包括所有子 节点。您可以将代理程序定义为多实例代理程序;在这种情况下,可以启动并停止单个实例,让其他实例保 持运行。

除子节点中的数据集以外,代理程序还可以定义代理程序级别的数据集,它们位于子节点外部。

在 Tivoli Enterprise Portal 的导航器树中,子节点类型显示在代理程序名称下,子节点实例显示在子节点类型下。与代理程序一样,子节点由受管系统名称 (MSN) 识别,例如 94:Hill.cmn。

例如, 在<u>第 160 页的图 20</u> 所示的导航器树中, Watching Over Our Friends 是有三个资源(Boarders、 Common Areas 和 Kennel Runs)以及两种子节点类型(Common Area 和 Kennel Run)的代理程序。其 中两个资源(Common Area 和 Kennel Run)具有针对其定义的子节点类型。第三个资源(Boarder)不需 要子节点,它由表中基本代理程序级别的单一行表示。Common Area 子节点类型有三个子节点实例: 94:Hill:cmn、94:Meadow:cmn 和 94:Tree:cmn,它们表示内核中的三个公共区域。Kennel Run 子节 点类型有四个子节点实例:94:system1:run、94:system2:run、94:system4:run 和 94:system5:run,它们表示四个 Kennel Run。



#### 图 20. 导航器树中的子节点

单个代理程序可以两种方式使用子节点:

- ·代理程序可以有类型相同的不同子节点。
- ·代理程序可以有类型不同的子节点。

#### 用于来自不同来源的同一数据的子节点

您可以使用同一类型的子节点来表示受监视资源类型的多个实例。同一类型的每个子节点包含相同的属性组和特定受监视资源实例的正确值。子节点的数量随代理程序配置不同而异。<u>第161页的图21</u>中的示例显示不同系统的监视。



图 21. 监视不同系统的子节点

## 用于多类数据的子节点

当一个代理程序监视多种类型的受监视资源时,您可以为每种资源类型创建一种子节点类型。每个子节点都 会包含该子节点类型中定义的信息。以下示例显示两种子节点类型。每种类型都监视一种不同类型的资源, 并针对每种资源提供不同类型的数据:

#### · Common Area

· Kennel Run

<u>第162页的图22</u>中的代理程序运行每种子节点类型的一个副本。特定代理程序可以创建已定义的代理程序的任意子集。子节点可以用于模拟 Tivoli Monitoring V5 概要文件。



图 22. 导航器树中的子节点类型

这两种使用子节点的方法可以在同一代理程序中使用,其中每种类型都可以有多个子节点实例。

<u>第162</u>页的图 22 显示监视以下两类资源的两类子节点: Common Area 和 Kennel Run。另外,还为每种类型定义了若干子节点。Common Area 类型的有三个子节点;这些子节点的标识如下: Meadow、Hill 和 Tree。另外还有四个 Kennel 类型的子节点,它们分别从专用于 Kennel Run 的不同系统收集数据;这些子节 点的标识如下: system1、system2、system4 和 system5。

**注:** 子节点标识的前 24 个字符对于 IBM Tivoli Monitoring 安装中该子节点类型的所有实例而言必须是唯一的。

#### 子节点中的数据提供程序

子节点可以包含来自不同数据提供程序类型的任意混合数据。可以在子节点中使用最新的 Agent Builder 数据提供程序,包括以下数据提供程序:

- $\cdot WMI$
- · Perfmon
- ·Windows 事件日志
- · SNMP
- ·SNMP 事件
- ·JMX
- · ICMP Ping
- ·脚本
- ·日志
- · CIM
- ·JDBC

· HTTP

- · SOAP
- ·套接字
- $\cdot$  Java API

子节点还可以包含连接属性组,用于组合来自相同子节点的两个其他属性组或代理程序级别属性组的数据。

#### 子节点状态

有两种方式可用来确定子节点代理程序的状态。第一种方式是查看"性能对象状态"属性组中显示的数据。 该属性组显示了代理程序中同一级别上的其他所有属性组的状态。代理程序级别的"性能对象状态"属性组 显示了代理程序级别上的其他属性组的收集状态。而每个子节点中的"性能对象状态"属性组显示了该子节 点中的属性组的收集状态。

Agent Builder 还将为每个子节点类型创建一个属性组,该属性组为该类型的每个已配置子节点显示一行。在 (<u>第 163 页的图 23</u>)中的示例中,四个节点正在运行以收集数据。



图 23. 监视相同子节点类型的多个子节点实例

在 IBM Tivoli Monitoring 环境中, 性能对象子节点子节点包含导航器树中可见的数据,并具有一些用于监视 其他数据收集状态的情境。

在 IBM Cloud Application Performance Management 环境中,您可以创建阈值来监视性能对象状态数据。

<u>第164页的图24</u>中的示例显示数据收集失败的情况(由于找不到脚本 shell 命令而失败)。通常除 NO\_ERROR 外的所有值都表示存在问题。对于子节点中定义的每个数据收集器,表中都有对应的一行。



图 24. 示例:子节点中的数据收集

#### IBM Cloud Application Performance Management 中的子节点

在 IBM Cloud Application Performance Management 中,您可以将代理程序实例和/或子节点实例定义为受监视资源,每个资源对应于一个摘要仪表板。

子节点仪表板无法显示代理程序级别的数据。要在此环境中显示代理程序级别的数据,请为代理程序定义摘 要仪表板。

根据所选设置,代理程序和子节点资源可以显示在同一级别,没有层次结构之分,或者将子节点资源列示为 代理程序资源的子代。

有关配置代理程序和子节点资源的指示信息,请参阅<u>第 187 页的『第 12 章 为 Cloud APM 准备代理程</u> <u>序』</u>。

# 创建子节点

您可以在创建或编辑代理程序时创建子节点。

#### 过程

- 1. 执行下列其中一个步骤:
  - · 使用"代理程序"向导创建新代理程序时,在"代理程序初始数据源"页面上,单击监视数据类别区域中的数据源分组。
  - · 对于现有代理程序,请在 Agent Editor 中执行以下步骤:
    - a. 单击数据源选项卡,以打开"数据源定义"页面。
    - b. 选择代理程序并单击添加至选定项。
c. 在"数据源位置"页面上的监视数据类别区域中,单击数据源分组。

- 2. 在数据源区域中,单击子节点定义。
- 3. 单击下一步。
- 4. 按如下所示填写"子节点信息"页面,以定义新的子节点:
  - a) 在名称字段中, 输入要创建的子节点的名称。
  - b) 在类型字段中, 输入1到3个字符(使用数字和/或字母), 以指示所创建子节点的类型。
  - c) 在描述字段中, 输入要创建的子节点的描述。
  - d) 单击**针对此类子节点显示节点属性组**复选框,以隐藏或显示可用性属性组。有关此属性组的更多详细 信息,请参阅<u>第 235 页的『可用性节点』</u>。
  - e) 单击下一步。
- 5. 填写"初始子节点数据源"页面,以选择一个数据源作为新子节点中的第一个项。在监视数据类别列表 中单击类别,并在数据源列表中单击数据源。然后,单击下一步。

提示:您可以照常创建数据源。另外,您也可以将已创建的一个或多个数据源移入导航器组。要移动数据 源,请单击**现有数据源**,并在"**当前定义的数据源"**页面中选择数据源。

要点:不能将进程、Windows 服务或命令返回码数据源包括在子节点中。作为变通方法,您可以编写脚本来确定必要的进程或服务信息,并使用脚本输出数据源。

6. 如果代理程序包含定制配置属性,或者所选数据源需要配置,请使用"**子节点配置覆盖**"页面来选择配置属性。

在"**子节点配置覆盖**"页面中,选择要在代理程序级别用于子节点的配置属性。然后,请选择要随每个 子节点而不同的配置属性。

使用移动、复制和移除,按照第166页的『配置子节点』所述指定配置属性。

7. 单击**下一步**。

"数据源定义"页面随即显示。

# 子节点配置

定义子节点类型时,将会针对该子节点专门定义一个配置节。

子节点配置节与其他配置节有几点不同:

- ·子节点节中的属性集可以复制,因此有多组属性。每组属性构成其自己的节。所有节的布局都相同,但可以在每个节中输入不同的值。
- 与此相反,其他节(即代理程序级别的节)中的属性在运行时配置期间仅显示一次。它们不会构成子节, 且无法复制或移除。

请参阅<u>第169页的『子节点配置示例』</u>,以获取配置子节点的 GUI 和命令行示例。

- ·对于在运行时配置期间创建的子节点节的每个副本,代理程序将创建单独的子节点实例。所有子节点实例 的类型相同。
- ·子节点节中的属性名可以与代理程序级别的节中的属性名重复。出现重复名称时,子节点属性值会覆盖代 理程序级别的属性值。
- · 在 IBM Tivoli Monitoring V6.2.1 和更高版本中,子节点节可以具有缺省属性值,应用于该类型的子节点的所有实例。此功能允许对单个属性值执行三层查找,如下所示:
- 1. 代理程序从子节点实例子节获取属性值。
- 2. 如果在子节点实例级别未配置任何值,那么将从子节点缺省级别获取属性值。
- 3. 如果在上述两个级别均未配置值,那么将从代理程序级别的节获取属性值。

请参阅<u>第169页的『子节点配置示例』</u>,以获取配置子节点的 GUI 和命令行示例。

# 配置子节点

使用子节点配置覆盖页面可配置子节点数据源。

## 开始之前

使用第164页的『创建子节点』中的步骤来创建子节点。

# 关于此任务

将数据源添加到子节点时,如果该数据源需要配置,那么会显示"**子节点配置覆盖**"页面。它会显示定制的 配置属性,以及任何其他适用于子节点类型的配置属性。

#### 过程

- · 在"**子节点配置覆盖**"窗口中,选择要在代理程序级别用于子节点的配置属性。另外,请选择要随每个 子节点而不同的配置属性。
- 使用复制>> 来复制配置属性,使它们同时存在于代理程序级别和子节点级别。
   代理程序先在子节点级别查找值,如果找不到,接着在代理程序级别上查找。如果某个同时存在于这两个级别的属性是必需属性,那么它只是在代理程序级别上必需,而在子节点级别是可选项。
- · 使用**移动 >>** 将属性从代理程序级别移至子节点级别。对于代理程序级别数据源或其他类型子节点所需的 属性,移动 >> 不可用。
- · 使用**移除**来移除两个列表中的一个。属性只有同时在代理程序级别和子节点级别列出时才可移除。此功 能不能用于完全移除某个属性。
- · 使用 << 复制将属性从子节点级别复制到代理程序级别。
- · 使用 << 移动将属性从子节点级别移至代理程序级别。

## 下一步做什么

您可使用 Agent Editor 更改现有子节点的配置。

# 子节点配置覆盖

使用"子节点配置覆盖"可将代理程序配置属性覆盖为特定于子节点的属性。

第166页的『配置子节点』中的过程描述如何针对自动生成的属性管理子节点配置。管理定制配置属性与此相似。所定义的任何定制配置属性都会显示在**子节点配置覆盖**窗口中。

在将定制属性从子节点级别复制或移动到代理程序级别时,系统会提示您指定要将其放入哪个节。您可以选择现有的定制节,也可以输入新定制节的名称。

#### 选择子节点配置属性

如果没有子节点,那么数据源类型的所有实例将共享配置参数。例如,所有 SNMP 属性组将使用相同的共用 名来连接到同一主机。如果有子节点,且在子节点级别指定 SNMP\_HOST 属性,那么子节点的各个实例可以 连接到不同的主机。

开发代理程序时,选择要在子节点级别覆盖的属性是一项重要的考虑事项。选择了过多属性时,子节点配置 节会变得杂乱,难以管理。如果选择的属性过少,那么用户希望某个属性因子节点不同而有变化时,代理程 序功能可能会受限。

以下属性无法复制到子节点级别。(所有子节点及基本代理程序中的所有属性组都必须使用相同的 SNMP 版本和 JMX 连接类型):

- ·SNMP 版本
- ·JMX MBean Server 连接类型
- ·Java 主目录
- ·Java 跟踪级别

- · JVM 自变量
- ·外部 JAR 文件的类路径
- ·套接字数据源端口号
- ·JMX 或 JDBC 类路径设置

# 高级子节点配置

使用高级子节点配置可以在子节点中覆盖代理程序配置属性。

# 关于此任务

在 IBM Tivoli Monitoring V6.2.1 和更高版本代理程序中有一个选项,启用此选项即可在子节点实例中覆盖来 自任意代理程序级别配置节的属性。在"**子节点配置覆盖**"页面上,有一个标注为**允许在任何子节点中覆盖** 任何配置属性的复选框。有关更多信息,请参阅<u>第 166 页的『子节点配置覆盖</u>』。要启用此选项,在对代理 程序命名(<u>第 13 页的『对代理程序命名和配置</u>』)时,必须选择 6.2.1 作为最低 ITM 版本。如果选择此选 项,那么每个子节点实例都可以覆盖任何代理程序级别配置节的任何属性。但是,只能从 GUI 覆盖此属性, 而无法从 itmcmd 命令行执行覆盖。

# 过程

**允许在任何子节点中覆盖任何配置属性**选项会在每个子节点配置面板上显示**高级**字段,其中包含一个列表。 高级字段中的初始选择提供了简要说明:请选择一个节以覆盖值。

- · 单击该列表时, 它会列出所有包含配置属性的非子节点节。
- · 选择一个节。

该节的属性将临时添加到子节点面板中。所更改的任何属性值将添加到针对该子节点定义的属性集。子 节点中的数据源会先在子节点中查找属性值,然后再从代理程序级别的节中查找。.

👙 Agent Configuration								
☞ SNMP Connection ☞ SNMP Version	Data about each	Data about each kennel run						
In Some Version 1 In Java	Kennel Run			^				
WebSphere Application Serve	These are initial explicitly change	property values for ne d in a section.	w sections. They will apply until a p	roperty value				
Common Area	ID							
	SNMP host	NMP host						
	Some Subno	Some Subnode Property						
	Advanced	- Select a section to	override values -					
	Kennel Run							
			Delete					
	Kennel Run	7						
	ID							
	SNMP host							
	Some Subno	de Property						
	Advanced		SNMP Version 1					
	SNMP comm	unity name	*					
	Confirm SNMF	community name	*					
<	<		Ш	>				
		Back	Next Home OK	Cancel				

图 25. 展开的 SNMP V1 属性

下列进一步的信息适用于覆盖来自代理程序级别节的属性:

- 在高级列表中选择代理程序级别的节时,不会显示之前复制到子节点节的属性。例如,在<u>第168页的</u>
   图 25 中, SNMP 主机不会显示在高级列表之后,因为它已复制到子节点属性,并已显示。
- 在高级列表中,不可选择不包含可覆盖属性的节。
- 系统会保留您为一个节输入的覆盖值,即使您选择另一个节以显示其他属性也是如此。
- 选中**允许在任何子节点中覆盖任何配置属性**,以便在代理程序中启用此功能。

# 从命令行配置子节点

在 IBM Tivoli Monitoring 环境中,还可以使用命令行来配置子节点。

# 开始之前

有关子节点配置的更多信息,请参阅<u>第165页的『子节点配置』</u>。

# 关于此任务

# 过程

· 要从命令行配置子节点实例, 请使用以下命令:

tacmd configureSystem -m HOSTNAME:00 -p
section\_name:subnode\_instance\_id.property\_name=value

其中:

#### section\_name 上乙共占米刑t

与子节点类型相同

subnode\_instance\_id 配置期间定义的子节点标识。

# property\_name

配置属性的名称

# value

属性的值

# 子节点配置示例

本节说明如何配置具有一个已定义的子节点的样本代理程序。

## 示例:

此示例说明如何配置一个样本代理程序,它有一个名为 Example Subnode 且类型为 exs 的子节点以及下列 三个配置属性:

- ·Agent Cfg(实际属性名为K00\_AGENT\_CFG)仅在代理程序级别定义。
- ·Subnode Cfg(实际属性名为 K00\_SUBNODE\_CFG) 仅在 Example Subnode 中定义。
- ·Overridable Cfg(实际属性名为KOO\_OVERRIDABLE\_CFG)在代理程序级别定义,并已复制到 Example Subnode。

第 169 页的图 26 显示 Agent Editor 中 "运行时配置信息"页面上的这些配置属性。

😓 Agent Editor Example Project 🛛 🔀	
Runtime Configuration Information	ନ୍ଧ
Runtime Configuration Information	
<ul> <li>□ <sup>™</sup> Custom Configuration</li> <li>□ <sup>™</sup> Top</li> <li><sup>™</sup> Agent Cfg</li> <li>□ <sup>™</sup> Main</li> <li><sup>™</sup> Overridable Cfg</li> <li>□ <sup>™</sup> Subnode configuration</li> <li>□ <sup>™</sup> Example Subnode</li> <li><sup>™</sup> Subnode Cfg</li> <li><sup>™</sup> Overridable Cfg</li> </ul>	Add
Format configuration sections as wizard pages	
Runtime Configuration Details	
Information about the configuration section	
Label Example Subnode	
Description	
Subnode Configuration Overrides	
Agent Information Data Sources Runtime Configuration itm_toolkit_agent.xml	

#### 图 26. Agent Builder 中的配置属性定义

配置此示例代理程序时,显示的第一个页面是 **Top** 节,其中包含<u>第 170 页的图 27</u> 所示的 **Agent Cfg** 属性。 此属性是代理程序级别的属性,因此在代理程序配置期间会显示一次。Example Subnode 的任何实例都可以 看到该属性值,但所有实例看到的是同一个值。

👙 Agent Configuration				
<ul> <li>Top</li> <li>Main</li> <li>Example Subnode</li> </ul>	Agent Cfg	a value		
	Back Ne	xt Home	ОК	Cancel

# 图 27. 包含 Agent Cfg 属性的代理程序级别配置的 Top 节

如果要从 Tivoli Enterprise Monitoring Server 命令行进行配置,可以使用以下命令来设置 Agent Cfg 属性:

tacmd configureSystem -m HOSTNAME:00 -p "TOP.K00\_AGENT\_CFG=a value"

显示的下一个节是<u>第 171 页的图 28</u>中所示的 Main 节。这也是代理程序级别的节,其中包含代理程序级别的 Overridable Cfg 属性。此属性不同于 Agent Cfg 属性,它已复制到 Agent Builder 中的 Example Subnode。这表示可以在 Main 页面上输入此属性的缺省值。但是,任何 Example Subnode 实例都可以将此处输入的值覆盖为另一个值。

👙 Agent Configuration 🛛 🔀						
🗹 Τορ	Main configuration prope	rties				
<ul> <li>Main</li> <li>Example Subnode</li> </ul>	Overridable Cfg	default value				
	Back Next	Home OK Cancel				

# 图 28. 包含 Overridable Cfg 属性的代理程序范围缺省值的 Main 节

如果要从 Tivoli Enterprise Monitoring Server 命令行进行配置,可以使用以下命令设置此属性:

tacmd configureSystem -m HOSTNAME:00 -p "MAIN.K00\_OVERRIDABLE\_CFG=default value"

可以将这两个属性放入同一个代理程序级别的节中。您可以决定要创建多少个代理程序级别的定制节,以及如何将定制属性分配到这些节中。

显示的下一个节是<u>第 172 页的图 29</u>中所示的 Example Subnode 节。这是首次配置此代理程序,因此没有 已定义的子节点实例,也没有显示子节点实例子节。将会显示初始属性值子节,尽管它是可选项,而且某些 子节点类型可能不显示该子节。由于显示了初始属性值子节,因此可以输入任何配置属性的缺省值。 Overridable Cfg 属性已具有缺省值,该值获取自代理程序级别的同名属性。

👙 Agent Configuration	
ថ Top ថ Main	<u>N</u> ew
Example Subnode	Example Subnode These are initial property values for new sections. They will apply until a property value is explicitly changed in a section. Subnode Cfg Overridable Cfg default value Advanced • Select a section to override values • •
	Back Next Home OK Cancel

图 29. 没有子节点的 Example Subnode 节页面

通过在空的 Example Subnode 节页面(<u>第 173 页的图 30</u>)上执行以下操作,可以定义子节点实例:

- 1. 在初始 **Example Subnode** 节中的 **Subnode Cfg** 字段中,为该属性输入以下缺省字符串: sub-default value。
- 2. 单击新建。Example Subnode 子节随即显示在初始属性子节之后。
- 3. 在 Example Subnode 字段中,输入以下子节点实例标识: do。
- 4. 单击新建。另一个 Example Subnode 子节随即显示在第一个子节之后。
- 5. 在第二个 Example Subnode 字段中,输入以下子节点实例标识: re。
- 6. 在 Subnode Cfg 字段中,为 Subnode Cfg 属性输入以下值: sc override。
- 7. 在 Overridable Cfg 字段中,为 Overridable Cfg 属性输入以下值: oc override。

👙 Agent Configuration						
⊠ Top ⊠ Main	<u>N</u> ew					
Example Subnode	Example Subnode					
	These are initial property values for new sections. They will apply until a property value is explicitly changed in a section.					
	Subnode Cfg sub-default value					
	Overridable Cfg default value					
	Advanced - Select a section to override values -					
	Example Subnode					
	Delete					
	Example Subnode 🕐 do					
	Subnode Cfg sub-default value					
	Overridable Cfg default value					
	Advanced - Select a section to override values - 💌					
	Example Subnode					
	Delete					
	Example Subnode 🕐 re					
	Subnode Cfg sc override					
	Overridable Cfg oc override					
	Advanced - Select a section to override values -					
	Back Next Home OK Cancel					

图 30. 定义了两个子节点实例的 Example Subnode 节页面

两个新的子节会使得代理程序在启动时创建两个子节点实例。由于 do 子节点子节的属性未更改,因此该子 节点实例会使用缺省属性值。因为在 re 子节为属性输入了不同的值, re 子节点实例会使用输入的那些值。

您可以从 Tivoli Enterprise Monitoring Server 命令行使用以下命令来设置缺省值:

tacmd configureSystem -m HOSTNAME:00 -p "exs.K00\_SUBNODE\_CFG=sub-default value"

除了节名称用来识别子节点节之外,设置子节点缺省值的格式与设置代理程序级别属性的格式完全相同。

您可以从 Tivoli Enterprise Monitoring Server 命令行使用以下命令来创建子节点实例:

tacmd configureSystem -m HOSTNAME:00 -p "exs:do.K00\_OVERRIDABLE\_CFG=default value" \
 "exs:re.K00\_SUBNODE\_CFG=sc override" "exs:re.K00\_OVERRIDABLE\_CFG=oc override"

子节点实例标识将插入到节名称与属性名之间。使用命令行创建子节点实例时,即使所有属性都使用缺省值,也必须至少指定一个属性。否则,定义子节点实例时,无需在命令行上指定缺省值。

可以在一条命令中设置所有的代理程序配置属性。以下命令等同于上述所有命令的总和:

tacmd configureSystem -m HOSTNAME:00 -p "TOP.K00\_AGENT\_CFG=a value" \
 "MAIN.K00\_OVERRIDABLE\_CFG=default value" \
 "exs.K00\_SUBNODE\_CFG=sub-default value" \

# 子节点与 Windows 数据源

选择是否将 Windows 远程连接属性包括在代理程序中。

# 关于此任务

如果代理程序在代理程序级别有 Windows 数据源,而在子节点中没有这些数据源,那么在代理程序中包括 Windows 远程连接配置属性这一操作是可选的。Windows 数据源包括 Windows 事件日志、Windows Management Instrumentation 和 Windows 性能监视器。如果未包括配置属性,那么在缺省情况下,这些数 据源会监视本地 Windows 系统,且无需进行任何配置。缺省情况下,不会在任何子节点中包括 Windows 数 据源。

要选择是否在代理程序中包括 Windows 远程连接属性,请执行以下步骤:

# 过程

- 1. 在 "Windows Management Instrumentation (WMI) 信息"页面上,当数据源属性显示时,单击全局 选项。在创建数据源时选择全局选项,或者从 Agent Editor 的 "数据源"页面选择此选项。
- 2. 在"全局 Windows 数据源选项"窗口中,如果要将这些属性包括在代理程序中,请选择包括 Windows 远程连接配置。

# 子节点和脚本数据源

在子节点脚本中访问子节点实例配置属性,就像在代理程序级别的脚本中访问一样。

脚本可以访问代理程序级别的所有配置属性,以及所有子节点实例配置属性。如果代理程序级别的属性在子节点级别被覆盖,那么脚本只能访问子节点级别的属性值。

# 第10章 定制代理程序配置

定制进程、日志文件和脚本数据源的配置。

## 开始之前

如果要向代理程序添加 SNMP、JMX、CIM、JDBC、HTTP 和 SOAP 数据源,请按如下章节所述配置这些数据源:

- · 第 61 页的『监视来自简单网络管理协议 (SNMP) 服务器的数据』
- · <u>第 70 页的『监视 Java 管理扩展 (JMX) MBean』</u>
- · 第86页的『监视来自公共信息模型 (CIM) 的数据』
- · 第 109 页的『监视来自 Java 数据库连接 (JDBC) 的数据』
- · 第 117 页的『监视 HTTP 可用性和响应时间』
- · 第 123 页的『监视来自 SOAP 或其他 HTTP 数据源的数据』

## 关于此任务

使用本任务可以定制进程、日志文件和脚本数据源的配置,以使代理程序能够访问其所监视的应用程序。

所有代理程序都必须先进行配置,然后才能启动。所有代理程序都必须具有基本配置信息,例如连接至 Tivoli Enterprise Monitoring Server 的方法。在许多情况下,代理程序必须掌握更多配置信息才能访问运行 该代理程序的系统所特有的信息。例如,如果必须知道某个软件产品的安装位置,那么可以添加配置属性来 提示输入该信息。您可能提示输入的信息还可以是用于访问某个界面的用户标识和密码。

定制配置由代理程序开发者定义。并非所有代理程序需要定制配置,但是定制配置可以用于数据收集的以下 方面:

- ·与"进程监视器"中的自变量匹配
- ·与"进程监视器"中的命令行匹配
- ·构成日志文件路径或名称
- ·在脚本中定义环境变量

注:某些数据源(例如 JMX 和 SNMP)会自动添加此配置。

注:当 Agent Builder 自动添加特定于数据源的配置时,将只添加此配置的英语版。

在数据源定义期间,如果代理程序需要某个数据收集区域的特定于系统的信息,那么将显示**插入属性**或**插入** 配置属性。

例如,在创建用于监视日志文件的属性组时,将显示插入配置属性。

## 过程

1. 单击插入配置属性以显示"配置属性"窗口。

2. 在"配置属性"窗口中,单击属性,然后单击添加。

注:最初,没有为代理程序定义任何配置属性。

- 3. 在"运行时配置属性"窗口中,填写下列字段:
  - a) 在"部分"区域中,填写下列字段:

标签

描述属性的文本

描述

(可选)属性的描述

b) 在"属性"区域中,填写下列字段:

标签

在代理程序配置面板中显示的文本、它标识必须输入的信息。

#### 环境变量

环境变量显示在**环境变量**字段中,当您在"标签"字段中输入内容时,环境变量将进行更新。 Agent Builder 通过产品代码和标签自动构造环境变量的名称。如果您想独立于标签来更改环境 变量,那么可以取消选中**匹配标签**。

#### 描述

对所定义属性的描述(可选)。

# 类型

要收集的信息类型,这可以是下列其中一个选项:

#### 字符串

适用于必须收集的任何字母信息(例如、安装位置、用户名和主机名)。

#### 密码

适用于任何在存储时必须加密的信息。除了提供数据加密之外,还使用星号遮盖您在文本框 中输入的数据。此外,您需要输入该信息两次,以验证数据。

#### 数字

适用于任何数字信息(例如端口号)。

#### 选项

适用于特定值的列表。此选项将启用"选项"表。您可以通过单击**添加**来定义特定的值。输入的值将作为一组选项显示在代理程序配置面板上,您只能从这组选项中选择一项。

#### 只读文本

在配置代理程序时显示文本,但不收集信息。

#### 分隔符

显示水平分隔符,但不收集信息。

#### 文件浏览器

收集文件名字符串。请单击浏览来浏览系统文件,以查找所需的文件。

#### 缺省值

(可选)指定运行时第一次配置代理程序时在配置面板中显示的值。如果您想使用与 Windows 缺省值不同的 UNIX/Linux 缺省值,请单击**多个值**。

在"配置属性缺省值"窗口中,指定要用于 Windows 系统以及 UNIX 和 Linux 系统的缺省值。

注:多缺省值支持是仅在 IBM Tivoli Monitoring V6.2.1 及更高版本中才受支持的功能。如果您的 代理程序与 IBM Tivoli Monitoring V6.2 兼容,那么会有一个提示来提醒您此项需求,并且您可 以取消,或者在启用 V6.2.1 兼容性的情况下继续。

必需

如果配置代理程序时用户必须输入值,请选中此字段。如果用户可以选择是否输入值,请取消选中此字段。

c) 要添加选项, 请单击添加。

4. 在"配置属性值"窗口中,请填写标签和值字段。

"标签"显示为其中一个选项。如果选中此选项,那么该值将成为属性值。

5. 单击确定。

新配置部分及属性会显示在"配置属性"窗口中的定制配置下。

- 6. 可选: 要将另一个属性添加至现有部分,请选择该部分或者该部分中的现有属性,然后单击**添加**。请在 "**配置属性**"窗口的运行时配置树中进行选择。
- 7. 请填写新属性的字段。(请填写步骤 第 175 页的『3』 中的那些字段。)
- 8. 单击确定。此时将选中您最近添加的属性。
- 9. 保留选择,或者选择要插入到日志文件名中的属性。
- 10. 单击确定。此属性将插入到日志文件名中。 然后,您可以继续执行向导,以完成定义日志文件属性组。

注:尽管配置属性是在日志文件名的上下文中进行定义,但可用于其他位置。例如,另一接受配置属性的位置为脚本数据源。这种灵活性意味着,您可以使用脚本变量 \$K00\_APPLICATION\_LOG\_FILE(如果产品代码为 K00)来访问配置元素文件信息的值。您还可以使用 Windows 批处理文件变量%K00\_APPLICATION\_LOG\_FILE%。

# 使用 Agent Editor 更改配置属性

使用 Agent Editor 更改代理程序的配置属性。

# 关于此任务

本任务提供有关使用 Agent Editor 来查看、添加和更改配置属性的信息。

# 过程

- 1. 单击运行时配置选项卡。
- 2. 选择配置节, 然后单击添加。

**添加**的工作方式与其在<u>第175页的『第10章 定制代理程序配置』</u>中的工作方式相同。没有**编辑**选项, 这是因为,选中配置节或属性后即可对其进行编辑。

- 3. 选择配置属性, 以显示"运行时配置详细信息"区域。
- 4. 在"运行时配置详细信息"区域中,编辑字段以配置属性。

# 配置 Windows 远程连接

关于配置 Windows 远程连接的信息

# 关于此任务

Windows Management Instrumentation (WMI)、 "Windows 性能监视器 (Perfmon)"及 "Windows 事件日志"数据源可以监视已安装代理程序的系统上的数据。这些数据源也可以监视远程 Windows 系统上的数据。这三种数据源类型称为 Windows 数据源。如果这些 Windows 数据源正在以远程方式监视数据,那么它们将共享定义这些数据源的代理程序级别的"Windows 远程连接"配置属性。

如果您在代理程序的基本级别定义 Windows 数据源,那么"Windows 远程连接"配置属性不会自动添加至 代理程序。为了保持与可能使用 Windows 数据提供者的先前版本代理程序的兼容性,在启用远程监视之 前,不会添加这些配置属性。代理程序中的 Windows 数据源监视已安装该代理程序的本地 Windows 系统上 的数据。

如果您在代理程序中的某个子节点中定义了 Windows 数据源,那么"Windows 远程连接"配置属性会自动 添加至该代理程序。如果 Windows 数据源是在子节点中,那么它必须支持"Windows 远程连接"。除非从 代理程序的所有子节点中移除所有 Windows 数据源,否则无法取消选中此选项。子节点的每个实例都可以 配置为监视不同的远程 Windows 系统。子节点中的所有 Windows 数据源共享"Windows 远程连接"配置 属性。

要配置基本代理程序,以便对单个远程 Windows 系统进行远程监视,请完成以下过程。

# 过程

1. 在 Agent Editor 的"数据源定义"窗口中,单击全局选项。

"全局 Windows 数据源选项"窗口随即打开。

- 2. 选择包括 Windows 远程连接配置。
- 3. 单击确定。

# 结果

在 Agent Editor 的"运行时配置信息"页面中,可以通过选择 Windows 远程访问配置 > Windows 远程连接访问下列特定于连接的配置属性。

## 远程 Windows 主机

远程 Windows 计算机的主机名

远程 Windows 密码

远程 Windows 的密码

远程 Windows 域\用户名

远程 Windows 主机的用户名

# 下一步做什么

您可以使用 Agent Editor 来查看、添加和更改配置属性。有关指示信息,请参阅<u>第 177 页的『使用 Agent Editor 更改配置属性』</u>。如果子节点中定义了 Windows 数据源,您还可以指定"子节点配置覆盖"。有关指示信息,请参阅<u>第 165 页的『子节点配置』</u>。

# 创建具有 Windows Management Instrumentation (WMI) 权限的用户

在 Windows 系统上,您可以添加并配置具有权限的用户,以允许进行 WMI 浏览。

# 关于此任务

如果代理程序通过使用 Windows Management Instrumentation (WMI) 从远程系统收集数据,那么它必须有 权访问远程系统上的 WMI 数据。您提供有权访问远程系统上 WMI 数据的帐户的凭证后,代理程序便能够访 问该系统上的 WMI 数据。此过程适用于 Windows 7、Windows 2008 Server 和 Windows Vista。

**注:** 代理程序还可以使用"Windows 性能监视器 (Perfmon)"和"Windows 事件日志"数据源来访问远程 Windows 系统上的数据。但是,对于"Windows 性能监视器 (Perfmon)"和"Windows 事件日志"数据 源,您必须提供远程系统的管理员凭证。

# 过程

- 1. 创建用户帐户:
  - a. 转至 Windows 开始 > 管理工具 > 计算机管理。"计算机管理"窗口随即打开。
  - b. 展开本地用户和组。
  - c. 右键单击用户文件夹并选择新用户。
  - d. 填写用户详细信息, 并单击创建和关闭。
- 2. 配置新用户帐户的组成员资格:
  - a. 在"计算机管理"窗口中,选择用户文件夹。
  - b. 右键单击新用户帐户, 然后选择属性。
  - c. 单击**隶属于**选项卡。
  - d. 单击添加。
  - e. 单击高级。
  - f. 单击立即查找。
  - g. 选择下列组:
    - · Distributed COM Users
    - · Performance Log Users
    - · Remote Desktop Users

提示:如果要选择多个组,请按住 Ctrl 键,然后单击。

- h. 单击确定, 直至返回到"计算机管理"窗口为止。
- i. 选择文件 > 退出以退出"计算机管理"窗口。
- 3. 分配分布式组件对象模型 (DCOM) 权限:
  - a. 转至 Windows 开始 > 管理工具 > 组件服务。"组件服务"窗口随即打开。

- b. 展开组件服务 > 计算机 > 我的电脑。
- c. 右键单击我的电脑并选择属性。"我的电脑属性"窗口随即打开。
- d. 单击 COM 安全选项卡。
- e. 在"访问权限"区域中, 单击编辑限制
- f. 在 Distributed COM Users 中,确认已选中本地访问和远程访问。
- g. 单击确定以保存设置。
- h. 在"我的电脑属性"窗口中的"启动和激活权限"区域中,单击编辑限制
- i. 在 Distributed COM Users 中,确认已选中本地启动、远程启动、本地激活和远程激活。
- j. 单击确定以保存设置, 然后再次单击确定以关闭"我的电脑属性"窗口。
- k. 选择文件 > 退出以退出"组件服务"窗口。
- 4. 配置 WMI 名称空间安全分配
  - a. 转至 Windows 开始 > 运行...。
  - b. 输入 wmimgmt.msc 并单击确定。
  - c. 右键单击 WMI 控件(本地)并选择属性。
  - d. 单击安全选项卡。
  - e. 单击安全设置。
  - f. 单击添加。
  - g. 单击高级。
  - h. 单击立即查找。
  - i. 选择新用户帐户,并单击确定,直至返回到"安全设置 Root"窗口为止。
  - j. 单击高级并选择新添加的用户帐户。
  - k. 单击**编辑**。
  - I. 从应用于:菜单选项中,选择这个名称空间和子名称空间。
  - m. 在执行方法中,确认已选中启用帐户、远程启用和读取安全性。
  - n. 单击确定, 直至返回到 wmimgmt 窗口为止。
  - o. 选择文件 > 退出以退出 wmimgmt 窗口。

#### 下一步做什么

有关从远程系统收集 WMI 数据的更多信息,请参阅<u>第 58 页的『监视来自 Windows Management Instrumentation (WMI)</u>的数据』。

# 配置安全 Shell (SSH) 远程连接

有关配置 SSH 远程连接的信息

#### 关于此任务

脚本数据源可以监视安装了代理程序的系统上的数据,还可以监视远程系统上的数据。如果这些脚本数据源 以远程方式监视数据,那么它们将共享定义这些数据源的代理程序级别的 SSH 远程连接配置属性。在启用远 程监视之前,代理程序的早期版本可能使用了脚本数据提供程序。为了保持与代理程序的早期版本的兼容 性,SSH 远程连接配置属性不会自动添加到代理程序。代理程序中的脚本数据源将监视安装了该代理程序的 本地系统上的数据。

如果您在子节点中定义脚本数据源,并选中了**允许使用 SSH 进行数据收集**,那么可以配置每个子节点实例 以监视不同的远程系统。子节点中的所有脚本数据源都将共享 SSH 远程连接配置属性。

如果要让代理程序以远程方式监视远程系统,请完成以下过程。

## 过程

在 Agent Editor 中脚本数据源的"数据源定义"窗口中,选中允许使用 SSH 进行数据收集。

### 结果

在 Agent Editor 的"运行时配置信息"页面中,可以通过选择 Secure Shell (SSH) 的配置 > SSH 远程连接 访问下列特定于连接的配置属性。

#### 网络地址

远程计算机的 IP 地址或主机名。

#### SSH 端口号

运行 SSH 服务器的 IP 端口号。缺省值是 22。

#### 认证类型

登录到远程 SSH 服务器时要使用的认证类型。您可以选择 Password 或 Public Key。

#### 在每个收集间隔后从远程系统中断开连接

此选项确定在收集数据后,脚本数据提供程序是否断开与远程系统的登录会话。在缺省情况下,值为 No。

#### 在每个收集间隔后从远程系统中移除脚本

此选项用于在每个数据收集间隔之后,从远程系统中删除脚本。在缺省情况下,值为 No。

如果"认证类型"设置为"密码",那么从 Agent Editor 的"运行时配置信息"页面中,可以通过选择 Secure Shell (SSH) 的配置 > 密码访问下列配置属性:

#### 用户名

远程系统的用户名称

#### 密码

远程系统的密码

如果"认证类型"设置为"公用密钥",那么从 Agent Editor 的"运行时配置信息"页面中,可以通过选择 Secure Shell (SSH) 的配置 > 公用密钥访问下列配置属性:

### 用户名

与公用密钥文件相关联的用户名

#### 公用密钥文件

与用户相关联的公用密钥文件

#### 专用密钥文件

与用户相关联的专用密钥文件

#### 密码

用于将专用密钥文件解锁的密码

## 下一步做什么

您可以使用 Agent Editor 来查看、添加和更改配置属性。有关指示信息,请参阅<u>第 177 页的『使用 Agent</u> Editor 更改配置属性』。如果 SSH 远程连接配置属性包含在子节点中,那么您还可以指定"子节点配置覆 盖"。有关指示信息,请参阅<u>第 165 页的『子节点配置』</u>。

# 第11章 创建工作空间、"执行操作"命令和情境

在 IBM Tivoli Monitoring 环境中安装代理程序之后,您可以为监视解决方案创建工作空间、查询、"执行操作"命令及情境。

您创建的情境、工作空间、"执行操作"命令和查询可以包括在安装程序包中。为了使情境、工作空间和代 理程序本身使用同一个安装映像,情境文件和工作空间文件必须与代理程序位于同一个项目中。Agent Builder 提供了在代理程序项目中创建适当文件的向导。有关导入应用程序支持文件的信息,请参阅<u>第 217</u> 页的『第 15 章 导入应用程序支持文件』。

# 创建情境、"执行操作"命令和查询

查找信息以帮助创建情境、"执行操作"命令和查询。

要创建情境、"执行操作"命令和查询,请使用 Tivoli Enterprise Portal 和嵌入式情境编辑器。有关如何创建情境的详细信息,请参阅《Tivoli Enterprise Portal 用户指南》。您还可以使用与 Tivoli Enterprise Portal Server 一起安装的帮助文档。Agent Builder 监视代理程序能够识别一组特定的"执行操作"命令,并对其执行特殊处理。有关这些特殊的"执行操作"命令的更多信息,请参阅<u>第 325 页的『附录 J "执行操作"命</u>令参考』。

系统监视器代理程序的情境的创建方式与使用 Tivoli Enterprise Portal 情境编辑器或 tacmd createSit 命 令创建的企业情境不同。对于系统监视器代理程序,将在代理程序的本地专用情境配置 XML 文件中创建专用 情境。有关为系统监视器代理程序创建情境的信息,请参阅 *IBM Tivoli Monitoring Administrator's Guide* 的 "Agent Autonomy" 一章中的"Private situations"。

# 创建工作空间

将 Tivoli Enterprise Portal 置于"管理员"方式,以创建可以导出并包括在解决方案中的工作空间。

#### 关于此任务

在使用工作空间的环境中构建工作空间。在构建工作空间时,请更改计算机的显示设置,以环境中通常使用 的最低分辨率来构建工作空间。在较高分辨率下构建工作空间可能会创建杂乱的视图,这些视图在较低分辨 率下无法正常使用。

要创建可以导出并包括在解决方案中的工作空间,必须将 Tivoli Enterprise Portal 置于"管理员"方式。要将 Tivoli Enterprise Portal 置于"管理员"方式,请完成下列步骤:

#### 过程

1. 转至 ITM\_INSTALL/CNP 目录并打开 cnp.bat 文件。

如果您使用的是缺省安装,那么该目录为 C:\IBM\ITM\CNP。在 cnp.bat 文件中,必须更新 set \_CMD= %\_JAVA\_CMD% 行以包括选项 -Dcnp.candle.mode="\$\_KCJ\_\$"。

如果您想要在 Linux 或 AIX 系统上创建扩展,请使用以下路径:

/opt/IBM/ITM/*li263*/cj/bin/cnp.sh

其中, li263 是运行 Tivoli Enterprise Portal 的操作系统。

更新后的 set\_CMD=%\_JAVA\_CMD % 与以下示例类似:

set \_CMD= %\_JAVA\_CMD% -Dcnp.candle.mode="\$\_KCJ\_\$" -Xms64m -Xmx256m -showversion -noverify -classpath %CPATH% -Dkjr.trace.mode=LOCAL -Dkjr.trace.file=C:\IBM\ITM\CNP\LOGS\kcjras1.log -Dkjr.trace.params=ERROR -DORBtcpNoDelay=true -Dibm.stream.nio=true

-Dcnp.http.url.host=SKINANE -Dvbroker.agent.enableLocator=false -Dnv\_inst\_flag=%NV\_INST\_FLAG

<sup>-</sup>Dice.net.maxPersistentConnections=16 -Dice.net.persistentConnectionTimeout=1

% -Dnvwc.cwd=%NVWC\_WORKING\_DIR% -Dnvwc.java=%NVWC\_JAVA% candle.fw.pres.CMWApplet

注:在此处,此命令仅因为格式编排原因而显示为多行。

- 2. 打开新的 Tivoli Enterprise Portal Client,并以 sysadmin 用户标识登录。
- 3. 将 "sysadmin" 用户标识置于"管理员"方式。在 Tivoli Enterprise Portal 中,选择编辑 > 管理用户。选择 sysadmin,然后在许可权选项卡下,选择工作空间管理。选中工作空间管理方式复选框。 如果您正确地进行选择,那么桌面标题栏中将显示"\*管理方式\*"。



图 31. 设置 sysadmin 用户标识



图 32. 设置 sysadmin 用户标识(续)

Enterprise Status - SKINANE - SYSADMIN *ADMIN MODE*					
File Edit View Help					
(+ = → = 1 □ □ □ □ □ 10 13 (+ = → = 1)	3 4 🚳	🖽 🐼 🖬 🚱 🖽	🔛 🕅 🖻 💹 💬	🔮 🖅 🐚 🔟	
KE View: Physical View: Physical Stuaton Ex	ent Console				
	🗠 🯤 🐽	🔯 🛛 🔟 Total Even	ts: 10 Item Filter: Ente	rprise	
Prerprise Stat	us l	Situation Name	Display tem	Source	Impac
Windows Systems	TEST A	PP UP	NetCool SSM Agent	skinane/RESET_EXAMPLE	10 AVAILABL
	TEST A	PP UP	Net Config Process	skinane/BESET_EXAMPLE	00 De AVAILARI
Operation of the local sector of the local sec	TEST A	PP UP	Net Cool Service	skinane:RESET_EXAMPLE	00 WANDABI
A Ope	NT Log	Space Low	System	Primary:SKINANE:NT	System
	n NT Log	Space Low	Security	Primary:SKINANE:NT	System
	NT_Log	Space Low	Application	Primary:SKINANE:NT	System
E A Ope	n Scott Ev	ent Log		Primary SKINANE:NT	System
CO Ope	NT_Phys	sical Disk Busy Critical	0 C;	Primary:SKINANE:NT	Disk.
E Ope	NT_Phys	sical_Disk_Busy_Critical	_Total	Primary:SKINANE:NT	Disk
	scott 1	Field		Primary:SKINANE:NT	System -
Open Stuation Counts - Last 24 Hours	080	× E Message Log		-	080×
<b>B</b>		Status	Name	Display Item	Origi
		- 🧶 Open - Scot	tt_Event_Log		Primary SKINA +
		Cipen Scot	tt_1_Field		Primary SKINA
TEST APP UP		Open Scot	t_Event_Log		Primary SkiNA
	1	Copen Scol	IL_1_Field Obvioel Diel, Dues Cr	iteel Valat	Primary SKINA
Scott_Event_Log	5	Copen NT_	Physical_Disk_Busy_Ci Division Disk_Busy_Ci	tical _roat	Primary.SKINA
2		Copen nu	Filysical_Disk_Busy_Ci	nical o.c.	Drimmer CVINA
Scott_1_Field		Cosen Scat			Primary SKINA Primary SkINA
		Copen Scot	t_Event_Log tt 1 Field		Primary/SKINA Primary/SKINA Primary/SKINA
	Count	Open Scat     Open Scat     Open Scat	t_Event_Log tt_1_Field tt 1_Field		Primary SKINA Primary SKINA Primary SKINA Primary SKINA
NT_Physical_Disk_Busy_Citical	Count	Open Scot     Open Scot     Open Scot     Open Scot     Open Scot     Open Scot	t_Event_Log t_1_Field t_1_Field t_Event_Log		Primary SKINA Primary SKINA Primary SKINA Primary SKINA Primary SKINA
NT_Physical_Disk_Busy_Citical	Count	Open Scale     Open TES	t_event_Log t_1_Field t_1_Field t_Event_Log T_APP_UP	Net Cool Service	Primary, SKINA Primary, SKINA Primary, SKINA Primary, SKINA Primary, SKINA skinane, RESE
NT_Physical_Disk_Busy_Citical NT_Log_Space_Low	Count	Open Scale     Open Scale     Open Scale     Open Scale     Open Scale     Open Scale     Open TES     Open TES	t_1_Field t_1_Field t_1_Field t_Event_Log T_APP_UP T_APP_UP	Net Cool Service Net Config Process	Primary, SKINA Primary, SKINA Primary, SKINA Primary, SKINA Primary, SKINA skinane, RESE skinane, RESE
NT_Physical_Dide_Busy_Citical	Count	Open Scot     Open Scot     Open Scot     Open Scot     Open Scot     Open TES     Open TES     Open TES	T_Prent_Log t_1_Field t_1_Field t_1_Prent_Log T_APP_UP T_APP_UP T_APP_UP	Net Cool Service Net Config Process NetCool SSM Agent	Primary SKINA Primary SKINA Primary SKINA Primary SKINA Primary SKINA skinane RESE skinane RESE skinane RESE
NT_Physical_Dide_Buoy_Critical NT_Log_Space_Low MS_CRIne	Count	Open Scot Open Scot Open Scot Open Scot Open Scot Open TES Open TES Open TES Open Scot	t_event_Log t_1_Field t_1_Field t_Event_Log T_APP_UP T_APP_UP t_Event_Log	Net Cool Service Net Config Process NetCool SSM Agent	Primary SKINA Primary SKINA Primary SKINA Primary SKINA Primary SKINA skinane RESE skinane RESE Primary SKINA
NT_Physical_Disk_Busy_Citical NT_Log_Space_Low MS_Cotine	Count	Coen Sco Coen Sco Coen Sco Coen Sco Coen TES Coen TES Coen TES Coen Sco Coen Sco Coen Sco	Levent_Log tt_1_Field tt_1_Field tt_Revent_Log T_APP_UP T_APP_UP T_APP_UP tt_Prent_Log	Net Cool Service Net Config Process NetCool SSM Agent	Primary SKINA Primary SKINA Primary SKINA Primary SKINA Primary SKINA skinane RESE skinane RESE Primary SKINA *

图 33. 设置 sysadmin 用户标识(续)

### 下一步做什么

按<u>第 184 页的图 33</u>中的描述进入"管理员"方式之后,您现已准备好为应用程序创建工作空间。有关如何 定制和创建工作空间的信息,请参阅《<u>Tivoli Enterprise Portal 用户指南</u>》。另外,也可以使用与 Tivoli Enterprise Portal 组件一起安装的帮助文档。

如果您希望工作空间处于"只读"状态,而不可由客户删除,请为每个工作空间设置"不可编辑"和"不可删除"属性。在工作空间属性中,您必须选择以下属性:

# ·不允许修改

## ·IBM 提供的产品(标记为不可删除)

通过查看工作空间或单击具有控件的图标可转至这些属性。您也可以转至某个视图属性页面,然后转至属性 树中的工作空间级别。如果每个导航器项具有多个工作空间,请记住设置用于每个工作空间的属性。如以下 示例截屏所示:



图 34. 设置工作空间属性

> Properties - AVAILABILITY					
AVAILABILITY	-Workspace Io Name: Description:	AVAILABILITY			
1 Ber Chart 1 Ber Chart,1 1 Virtuel and Working Set Size	Workspace Options  Assign as default for this Navigator item  Do not allow modifications  Only selectable as the target of a Workspace Link				
	Product.s	rovided by EM (mark as non-deletable)			
		OK Cancel Apply Test Help			

图 35. 设置工作空间属性(续)

# 第12章为 Cloud APM 准备代理程序

如果要将代理程序与 IBM Cloud Application Performance Management 配合使用,您需要使用"**仪表板设** 置"向导来准备该代理程序。此向导将配置信息,您可在 Cloud APM 中的摘要仪表板和详细信息仪表板中查 看这些信息。它还设置 Cloud APM 对代理程序所必需的资源信息。

### 开始之前

为了成功地为 Cloud APM 准备代理程序,您需要确保该代理程序提供下列数据:

·一个或多个生成一行数据的数据集(属性组)。您可使用这些数据集中的属性来填充摘要仪表板。

**要点:**要在摘要仪表板中包含任何信息,需要在产生单行数据的数据集中提供该信息。一些数据源创建产 生多行数据的数据集;例如,进程、Windows 服务和命令返回码数据源会将数据放入产生多行的单个"可 用性"数据集中。在此类情况下,需要创建产生单行的已过滤数据集,以便将该数据包含在摘要仪表板 中。有关指示信息,请参阅第 155 页的『创建过滤属性组』。

- ·其中一个数据集中的数字属性,用于指示受监视服务的状态(正常、警告、严重或其他类似状态值)。您 必须为这个属性定义状态严重性值。有关定义状态严重性值的指示信息,请参阅<u>第38页的『指定用作状</u>态指示符的特性的严重性』。
- ·如果受监视应用程序用来提供服务的端口号固定不变,那么您必须知道这个端口。如果这个端口在不同部 署之间可能有变化,那么其中一个生成一行数据的数据集必须包含指示此端口的数字字段。
- ·如果该代理程序可安装在一台主机上,以监视正在另一主机上运行的服务器,那么其中一个数据集中的某 个字符串属性需指示服务器 IP 地址。如果该代理程序始终监视其运行所在的主机,那么不需要这样的属 性。

**提示:** 如果有提供主机名的属性可供使用,那么您可使用 nameToIpAddress 函数为 IP 地址创建派生属 性。有关创建派生属性的信息,请参阅<u>第 33 页的『创建派生属性』</u>。有关此函数的信息,请参阅 <u>第 43</u> 页的『ipAddressToName』。

如果代理程序包含子节点,那么这些需求适用于要为其创建仪表板的每个子节点。

#### 关于此任务

Cloud APM 监视资源。资源对应于代理程序的实例,或有时对应于子节点。要定义资源,您需要提供应用于 受监视服务的资源类型名称、服务器名称、IP 地址和端口号。

Cloud APM 显示每个受监视资源的摘要仪表板。摘要仪表板包含状态指示器;使用这个指示器(通常显示为 绿色、黄色或红色,以指示正常、警告或严重状态),用户一眼就可以看到资源状态。同一个仪表板可包含 其他一些高级别运行状况度量。

在摘要仪表板上,数据显示为单个的项。因此,包含此数据的数据集必须仅生成一行。

针对代理程序,可以选择性低提供详细信息仪表板。用户单击摘要仪表板即可查看详细信息仪表板。详细信 息仪表板可以显示表,因此可以在该仪表板上使用来自任何数据集的数据。

您必须选择摘要仪表板及详细信息仪表板上显示的属性(包括状态指示器)。

**要点:**所选属性中的数据每分钟自动从代理程序传递至 Cloud APM 服务器一次。指定过多的数据可能会导致 网络、服务器或受监视主机过载。请仅选择所需的属性。例如,如果必须显示连接数据集或派生属性,请勿 同时指定源属性。

要点:除这些属性以外的数据不会传递至 Cloud APM。您在 Cloud APM 中无法查看或使用其他数据,但是在 代理程序级别监视的阈值除外。如果在阈值中使用其他数据,则可能无法在 Cloud APM 控制台中查看阈值状态。

# 过程

1. 从"代理程序信息"视图,单击仪表板链接。

2. 在仪表板组件下,选择在仪表板中显示代理程序组件。

提示:或者,如果要创建代理程序以专用于 IBM Tivoli Monitoring,可以选择此代理程序不存在仪表板。 在此情况下,请勿完成此过程的后续步骤。不能将此类代理程序安装在 Cloud APM 环境中。

- 3. 单击仪表板设置向导链接。
- 4. 如果代理程序包含子节点,定义代理程序和子节点资源在 Cloud APM 中的布局:
  - ·选择基本代理程序实例以将基本代理程序(子节点之外的数据)显示为资源。
  - · 对于每个子节点,选择子节点"name"实例以将此节点显示为资源。
  - · (可选)对于选定的任何子节点,选择显示为代理程序的子代。在此情况下,子节点资源会作为代 理程序资源下的子代显示在 Cloud APM 控制台中的列表中。

Cloud APM 会针对您选择的每个组件显示摘要及详细信息仪表板。

**要点:** 如果再次运行此向导并取消选择代理程序或子节点,那么不会自动移除此代理程序或子节点的资源。要移除这些资源,请在"大纲"视图中展开**资源**,选择要删除的资源,并按键盘上的 Delete 键。

5. 在**属性选择 - 状态**页面中,选择指示受监视服务的状态的属性。可以使用返回单一数据行的组中的数字属性。

提示: 或者, 如果不想在仪表板中显示状态, 取消选择为此代理程序提供状态。

- 6. 在同一个页面中,您可选择是否要在摘要及详细信息仪表板中显示其他数据:
  - · 要在摘要仪表板中显示其他高级别运行状况度量,请确保选中选择其他要在此代理程序的摘要信息 中显示的属性框。否则,请取消选中这个框。
  - · 要在详细信息仪表板中显示其他数据,请确保选中选择其他要在此代理程序的详细信息中显示的属性框。否则,请取消选中这个框。(通常,请选中这个框,因为需要详细信息仪表板来显示足够多的数据,以使监视代理程序有意义)。

单击下一步。

- 7. 如果您已选中选择其他要在此代理程序的摘要信息中显示的属性,请在属性选择 摘要页面中最多选择 4 个要包括在摘要仪表板中的其他属性。可以使用返回单一数据行的组中的属性。单击下一步。
- 8. 如果您已选中选择其他要在此代理程序的详细信息中显示的属性,请在属性选择 详细信息页面中选择 要包含在详细信息仪表板中的属性。代理程序中的所有属性均可用;为避免发生性能问题,请尽可能包 含最少的属性。单击下一步。
- 9. 在**资源类型**页面中,输入要监视的服务器类型,例如 Email server 或 SampleCo Database Server。单击下一步。
- 10. 在**属性选择 软件服务器名称**页面中,在**固定名称**字段中输入固定的软件服务器名称,或者选择代理程 序中提供软件服务器名称的属性。此名称会显示给此特定受监视实例的用户,例如 JBoss 应用程序服务 器实例的名称。单击**下一步**。

**要点:**在同一台受监视主机上,请勿执行两个或两个以上具有相同软件服务器名称的监视代理程序、代 理程序实例或子节点。如果代理程序具有实例或子节点,请确保针对每一个实例或子节点生成唯一的软 件服务器名称。如果两个不同的代理程序生成相同的软件服务器名称,请勿将其安装在同一台受监视主 机上。

- 11. 在**属性选择 IP 地址**页面中,选择代理程序中指定受监视服务器或应用程序使用的主要接口连接 IP 地址(并非主机名)的属性。例如,HTTP 服务器的 HTTP 连接或者数据库服务器的数据库客户机连接。 或者,选中使用代理程序的 IP 地址,以使用运行代理程序的主机的地址。单击下一步。
- **12**. 在**属性选择 端口**页面中,输入受监视应用程序用来提供服务的端口,或选择代理程序中指定这个端口的数字属性。单击完成。
- 13. 如果同时选择了代理程序和一个或多个子节点作为资源,请单击下一步来输入下一个组件(代理程序或 子节点)的仪表板和资源信息。如果下一步按钮处于禁用状态,表示您已输入所有必要组件的信息;请 单击完成来完成向导。

#### 结果

在受监视主机上安装代理程序时,可以在状态概述选项卡中查看摘要和详细信息仪表板。

**要点:**从安装代理程序到仪表板可用最多可能有 30 分钟延迟,尤其是首次在环境中安装此代理程序类型和版本的情况下。

单击这个代理程序的摘要仪表板,即可查看详细信息仪表板。缺省情况下,详细信息仪表板中的所有信息都 显示为表。

您可使用属性详细信息选项卡将这些信息的定制显示配置为表和图表。

# 第13章在Agent Builder 中测试代理程序

使用 Agent Builder 创建代理程序之后,可以在 Agent Builder 中测试该代理程序。

测试代理程序可确保您所期望的监视数据就是显示的数据。通过测试代理程序,您可以了解如何修改或调整 代理程序设置,以确保显示的数据有用且准确。

您可使用以下方法在 Agent Builder 中测试代理程序:

- 1. 首先,使用 Agent Builder 的属性组测试功能来测试各个属性组,每次测试一个。有关更多信息,请参阅 <u>第 191 页的『属性组测试』</u>。
- 2. 完成属性组测试之后,可以使用 Agent Builder 的代理程序测试功能来一起测试代理程序中所有的属性 组。有关更多信息,请参阅<u>第 194 页的『代理程序全面测试』</u>。

要点:在 Agent Builder 中测试代理程序时,您会看到以下数字属性的特殊值:

- ·-1: 一般错误
- ·-2: 缺少数据
- ·-3:没有值(例如,数据库返回 NULL)

# 属性组测试

您可以使用属性组测试来测试使用 Agent Builder 创建的代理程序的属性组,每次测试一个属性组。在完成 属性组定义之前,您可以测试多个属性组。例如,您可以在定义新代理程序的属性组时,从"IBM Tivoli Monitoring 代理程序向导"启动测试。另外,也可以在向现有代理程序添加属性组时,从"IBM Tivoli Monitoring 代理程序组件向导"启动测试。

# 开始之前

在开始测试属性组之前,您可以选择性地执行下列操作:

- ·设置属性组测试首选项。有关更多信息,请参阅<u>第192页的『属性组测试-首选项』</u>。
- ·设置环境变量、配置属性以及适用的 Java 信息。有关更多信息,请参阅<u>第 193 页的『属性组测试 配置』</u>。

# 关于此任务

对于大部分数据源, Agent Builder 支持属性组测试功能

#### 过程

- 可以通过以下方法启动测试过程:
  - 1. 在创建代理程序或属性组期间,单击相关数据源信息页面上的测试。
  - 2. 创建代理程序之后,在 Agent Editor 的"数据源定义"页面上选择属性组,然后单击测试。有关 Agent Editor 的更多信息,请参阅<u>第 17 页的『第 4 章 使用 Agent Editor 修改代理程序』</u>。

在上述任一步骤中单击**测试**之后, "属性组测试"窗口随即显示。对于不同的数据源, 此窗口有所不同。

对于大部分数据源, Agent Builder 支持属性组测试功能。

有关特定属性组的测试过程的更多信息,请参阅随后的"测试"章节:

- Windows Management Instrumentation (WMI): 有关 WMI 测试过程的更多信息, 请参阅<u>第 59 页的</u> 『测试 WMI 属性组』
- Windows Performance Monitor (Perfmon): 有关 Perfmon 测试过程的更多信息,请参阅<u>第 61 页的</u>
   <u>『测试 Perfmon 属性组』</u>
- 简单网络管理协议 (SNMP): 有关 SNMP 测试的更多信息, 请参阅<u>第 65 页的『测试 SNMP 属性组』</u>

- 简单网络管理协议 (SNMP) 事件发送方: 有关 SNMP 事件测试过程的更多信息, 请参阅<u>第 69 页的『测</u>试 SNMP 事件属性组』
- Java 管理扩展 (JMX): 有关 JMX 测试过程的更多信息,请参阅 第 86 页的『测试 JMX 属性组』
- 公共信息模型 (CIM): 有关 CIM 测试过程的更多信息, 请参阅 第 88 页的『测试 CIM 属性组』
- 日志文件: 有关日志文件测试过程的更多信息, 请参阅 第 96 页的『测试日志文件属性组』
- 脚本: 有关脚本测试过程的更多信息, 请参阅<u>第106页的『监视脚本输出的步骤』</u>
- Java 数据库连接 (JDBC): 有关 JDBC 测试过程的更多信息,请参阅第 114 页的『测试 JDBC 属性组』
- 因特网控制报文协议 (ICMP) Ping: 有关 ICMP 测试过程的更多信息,请参阅<u>第 117 页的『测试 Ping</u>属性组』
- 超文本传输协议 (HTTP) 可用性: 有关 HTTP 测试过程的更多信息, 请参阅<u>第 123 页的『测试 HTTP 属</u>性组』
- SOAP: 有关 SOAP 测试过程的更多信息,请参阅 第 130 页的『测试 SOAP 属性组』
- 传输控制协议 (TCP) 套接字: 有关套接字测试过程的更多信息, 请参阅<u>第 138 页的『测试套接字属性</u> 组』
- Java 应用程序编程接口 (API): 有关 Java API 测试过程的更多信息,请参阅<u>第 149 页的『测试 Java</u> <u>应用程序属性组』</u>

有些数据源不具备属性组测试功能,例如:

- 您可以使用 Agent Builder 浏览器来查看系统上的即时数据。例如,您可以查看当前正在系统上运行的 进程。此外,还可以查看安装在系统上的服务(Windows 服务),以及存在的 Windows 事件日志。
- 您在代理程序中没有或基本没有可执行的定制(AIX二进制日志、命令返回码)。
- 无法使用属性组测试功能来测试连接的属性组以及经过滤的属性组,因为这些组基于多个属性组。

注:

- 1. 使用代理程序全面测试功能,可以对使用属性组测试功能无法测试的数据源进行测试。有关代理程序 全面测试的更多信息,请参阅第194页的『代理程序全面测试』。
- 2. 在测试数据源时,单击**收集数据**后,数据可能完全不显示,或者可能不是第一次单击后的最新数据。 在此类情况下,再次单击**收集数据**即可显示当前数据。

・ 调试:

所测试的每个数据源都有一个测试目录,该目录由 Agent Builder 创建。此目录用于该数据源的测试运行时环境。对该数据源运行的测试的相关日志文件存储在该目录下。这些日志文件可以帮助您调试在测试 期间发现的问题。

注:

- 1. 您单击**启动代理程序**以及单击**停止代理程序**之后,测试日志文件的位置就会以状态消息的形式显示在 "测试"窗口中。
- 2. 关闭 Agent Builder 时,所有的测试数据源目录随即删除。

# 属性组测试 - 首选项

在测试属性组之前设置首选项。

#### 关于此任务

在开始测试属性组之前,您可以选择性地设置部分首选项,以确定测试期间处理属性的方式。

#### 过程

- 1. 从 Agent Builder 菜单栏选择窗口 > 首选项。
  - "首选项"窗口随即打开。
- 2. 选择 Agent Builder。

与测试属性组相关联的首选项随即显示:

#### 测试时显示"数据类型已更改"对话框

选中此选项时, Agent Builder 会提供对属性数据类型的更改建议。当属性的数据类型与该属性的测试所返回的数据不匹配时, Agent Builder 就会提供更改建议。例如, 对属性定义的字符串长度太短, 无法容纳测试所返回的值。在此示例中, Agent Builder 会建议重新定义属性, 增加其字符串长度。取消选中此选项时, Agent Builder 在测试期间不会检查或推荐数据类型。此选项在缺省情况下已选中。

## 创建的脚本或日志属性的最大数目

在此字段中输入的值确定 Agent Builder 在日志文件或脚本属性组的初始测试期间解析的最大属性数。缺省值为 25。

3. 完成设置首选项后,请单击确定以保存设置并关闭"首选项"窗口。 如果要复原缺省设置,请先单击复原缺省值,然后单击确定。

# 属性组测试 - 配置

在测试属性组之前,请设置环境变量、配置属性和 Java 信息。

#### 关于此任务

在开始测试属性组之前,您可从数据源的"测试"窗口选择性地设置环境变量、配置属性和适用的 Java 信息。Java 信息是配置数据的子集。某些环境变量具有用于属性组测试的特殊值,这些值在缺省情况下已设定。如需进一步了解具有用于属性组测试的特殊值的环境变量,请参阅第 197 页的『测试环境变量』。

#### 过程

1. 可选: 在数据源的"测试"窗口中, 单击设置环境。

"**环境变量**"窗口随即打开。填写完成后, "**环境变量**"窗口会列出运行测试期间使用的所有环境变量。"环境变量"窗口的初始视图包含代理程序中定义的任何现有环境变量。另外, 还会列出您从该代 理程序的先前测试中添加的任何环境变量。

- a) 单击添加或编辑可以添加或编辑个别变量。
- b) 单击移除可以移除个别变量, 单击复原缺省值可以复原缺省变量并移除所有其他变量。
- c) 单击确定以保存更改并返回至"测试"窗口。
- 2. 可选: 在数据源的"测试"窗口中,单击配置。"运行时配置"窗口随即打开。
  - a) 单击编辑代理程序配置,即可使用"配置属性"窗口添加配置属性或编辑现有的代理程序配置属性。
  - b) 选择一个配置属性, 然后单击编辑即可编辑与所要测试的属性组相关的现有配置属性。
  - c)选择一个配置属性,然后单击复原缺省值可将配置属性复原为它的缺省值。

**要点:** 如果 JMX 数据源连接到远程 WebSphere Application Server, 请确保安装本地 WebSphere Application Server, 且 Java 位置设置为该服务器所使用的 JRE。有关设置连接的详细信息,请参阅<u>第</u>70页的『监视 Java 管理扩展 (JMX) MBean』。

- 3. 单击确定以保存更改并返回至"测试"窗口。
- 4. 注: 可以针对下列类型的属性组设置 Java 信息:
  - ·Java 管理扩展 (JMX)
  - ·Java 数据库连接 (JDBC)
  - ·超文本传输协议 (HTTP) 可用性
  - · SOAP
  - ·Java 应用程序编程接口 (API)

Java 信息是步骤 第 193 页的『2』 中描述的配置数据的子集

可选: 在数据源的"测试"窗口中, 单击 Java 信息。

"Java 信息"窗口随即打开。

a) 输入 Java 信息。

例如,浏览至 Java 运行时环境 (JRE) 的位置或输入该位置,选择 Java 跟踪级别,或者输入 JVM 自 变量。

# 代理程序全面测试

使用代理程序全面测试可以同时测试代理程序的所有属性组。您还可以使用代理程序全面测试对使用属性组测试功能无法测试的数据源进行测试。

## 关于此任务

您可以使用代理程序全面测试以 IBM Tivoli Monitoring 中运行代理程序的方式来运行代理程序,而不需要安装 IBM Tivoli Monitoring。

**要点:** 在 Windows 系统上,如果要在 Agent Builder 中对代理程序运行全面测试(请参阅<u>第 194 页的『代理</u> 程序全面测试』),请确保在"代理程序信息"窗口中选择运行 Agent Builder 的 32 位版本的操作系统,即 32 位的 Windows。在 Linux 系统上,必须选择 64 位版本。

## 过程

- 1. 打开"代理程序测试"透视图:
  - a) 在 Agent Editor 中, 打开代理程序信息选项卡。
  - b) 单击测试代理程序。

# Test Agent

Test the agent without leaving the Agent Builder. The Agent Test perspective will open where the agent can be configured and started.

图 36. Agent Editor 的"代理程序信息"页面的"测试代理程序"部分。

或者,从 Agent Builder 菜单中选择窗口 > 打开透视图 > 其他,选择代理程序测试,然后单击确定。 这样将打开"代理程序测试"透视图(<u>第196页的图38</u>)。"代理程序测试"视图显示您在 Agent Editor 中打开的代理程序;您可以测试其中的任何代理程序。还会显示"属性组测试"视图;此视图初 始为空。"属性组测试"视图显示在代理程序运行时从所选属性组收集的数据。

ନ୍ଦ୍ର

提示:如果未在编辑任何代理程序,那么"代理程序测试"透视图为空。要填充该视图,请转至 IBM Tivoli Monitoring 透视图,并在 Agent Editor 中打开代理程序。在 Agent Editor 中打开代理程序后,请返回到"代理程序测试"透视图以测试该代理程序。

2. 可选: 在开始测试前, 配置环境变量和配置属性。

您可以通过两种方法从"代理程序测试"视图访问"环境变量"和"运行时配置"窗口:

- · 在"**代理程序测试**"视图中右键单击代理程序以打开选择菜单。您可以从菜单中选择**设置环境**,以 打开"**环境变量**"窗口。您可从菜单中选择**配置**,以打开"**运行时配置**"窗口。
- · 单击"**代理程序测试**"视图工具栏上的"视图菜单"图标 <sup>2</sup>,以访问与上一选项相同的**设置环境**和 **配置**菜单项。

有关使用"**环境变量**"和"运行时配置"窗口的更多信息,请参阅<u>第 191 页的『属性组测试』</u>。

要点:

- a. 在代理程序中, 将自动填入与测试的每个属性组相关的最近一组配置。
- b. 对于属性组测试和代理程序全面测试,部分环境变量可能具有不同的缺省值。有关将特殊值用于属性组测试的环境变量的更多信息,请参阅(<u>第197页的『测试环境变量』</u>)。
- c. 如果 JMX 数据源连接到远程 WebSphere Application Server, 请确保安装本地 WebSphere Application Server, 且 Java 位置设置为该服务器使用的 JRE。有关设置连接的详细信息,请参阅<u>第70 页的『监视 Java 管理扩展 (JMX) MBean』</u>。
- d. 在 Java API、JDBC、JMX、HTTP 或 SOAP 数据源中,您可以使用 Java > JVM 自变量设置来控制 代理程序跟踪日志记录。设置以下值:

-DJAVA\_TRACE\_MAX\_FILES=files -DJAVA\_TRACE\_MAX\_FILE\_SIZE=size

其中, files 是保留的最大跟踪日志文件数量(缺省值为 4), size 是以千字节为单位的最大日志文件 大小(缺省值为 5000)。例如,您可以设置以下值:

-DJAVA\_TRACE\_MAX\_FILES=7 -DJAVA\_TRACE\_MAX\_FILE\_SIZE=100

在此情况下,代理程序将100千字节写入第一个日志文件,然后切换到第二个日志文件,以此类推。在写入七个100千字节的日志文件之后,将覆盖第一个日志文件。

- e. 如果代理程序包含子节点,那么在已安装版本中,您可以为不同的子节点设置不同的配置值,并可以单独为基本代理程序属性组设置配置值。但是,在全面代理程序测试配置中,每个配置值只能设置一次;该设置应用于基本代理程序以及任何子节点。只能测试每个子节点的一个实例。
- 3. 在"**代理程序测试**"视图中,选择要测试的代理程序,然后单击 🕨 启动代理程序图标。

此时将显示一个窗口,表明代理程序正在启动。代理程序启动后,其属性组会作为代理程序的子代显示 在"**代理程序测试**"视图中。属性组通过"属性组"图标 🗍 指示。

用于提供代理程序相关信息的状态属性组(Performance Object Status、Thread Pool Status 和 Take Action Status)也会作为代理程序子代显示在"**代理程序测试**"视图中。状态属性组通过**i**信息图标指示。

您可以同时启动并运行多个代理程序。

当代理程序启动时, 🦲 停止代理程序图标变为可用。

如果代理程序包含子节点或导航器组,它们会作为节点显示在"**代理程序测试**"视图中。子节点定义显 示在代理程序下方。子节点实例节点显示在子节点定义节点下方。属性组和导航器组将显示在子节点实 例节点下方。例如:

□ Agent Test 🛛		$\gg$	$\overline{}$	
<ul> <li>Fastnet Agent</li> <li>Mizzen Agent</li> <li>Tuskar Agent</li> <li>Windows_Data_nodes</li> <li>Managed_URLs</li> <li>Performance_Object_Status</li> <li>Take_Action_Status</li> <li>Thread Pool Status</li> </ul>	Subnode instance information group			
<ul> <li>URL_Objects</li> <li>Windows Data</li> <li>TestSubnodeInstance</li> <li>1 210_Performance_Object_Status</li> <li>Win32_WinSAT</li> <li>TimeData</li> <li>Win32_UTCTime</li> <li>Win32_LocalTime</li> </ul>	<ul> <li>Subnode name</li> <li>Subnode instance</li> <li>Navigator group</li> </ul>			

图 37. 突出显示了示例子节点和导航器组的"代理程序测试"视图。

您可以右键单击"代理程序测试"视图中的任意节点,以访问编辑和停止代理程序之类的菜单选项。编辑将针对 Agent Editor 中的选定节点打开"数据源定义"。

**注:** 直到运行中的代理程序停止并重新启动后,您使用 Agent Editor 所做的更改才会显示在该代理程序中。

4. 在"代理程序测试"视图中,选择要测试的第一个属性组。

选择属性组后,将为所选属性组开始数据收集。如果收集需要一些时间,那么窗口将指示数据收集正在进行中。数据收集完成时,所收集的数据会显示在"**属性组测试**"视图中,例如:

<b>A</b>			-					
2 Agent lest - Tuskar Agent/thm_tookt_agent.xml - 184 Two Montoomg Agent Builder								
File Eult Navigate Search P	roject kun	IBM TIVOII MONICON	ng Agent Editor	window Help				
<u>    -                                   </u>	] 💊 🔹 ] 🔗	<ul> <li>■ 21 × 21 ×</li> </ul>	$\leftarrow \diamond \bullet \bullet$					
🖹 🗄 Agent Test 🗐 IBM Tivo	li Monitoring							
🗖 Agent Test 🕴 🔰 📮 🖉 🖓 🖓 🧶 Agent Editor Fastnet Ag 💈 🧏 Agent Editor Mizzen Age 💈 😓 Agent Editor Tuskar Age 🖏 🖓 🖓 🗄 🗄 🗄 Outline 🛱								
Fastnet Agent     Mizzen Agent     Tuskar Agent		Agent Info	nt Information					
URL_Objects		General						Self Describing Agent
🚺 Managed_URLs		This section de	efines the genera	l agent information.				
i Performance_Object_: i Thread_Pool_Status	Status	Service name	Monitoring agent	t for Tuskar Agent				Cognos Information     Data Sources
i Take_Action_Status		Product cod	le K02		Company identifi	er miket		Runtime Configuration
		Version	623		Agent identifier	K02	_	OSLC - Open Services for Lifecycle
		Patch level			Display, name	Tuskar Agent	_	
		E Support	multiple instance	s of this paget	bibpidy nume			
		i Suppore	multiple instance.	s or this agent	Minimum 11M ve	sion [6.2.1	-	
		Copyright	Copyright Mike	T Corp 2011. All rights	reserved		-	
		Agent Information	Data Sources	Runtime Configuration	itm_toolkit_agent.	xml		
🗖 Attribute Group Test 🛛								la ¬ ~ €
Data collection at 10-Sep-2012 1	11:33:46 retur	ned 3 data rows.						
URL	Response_T	ime Page_Size	Page_Objects	Total_Object_Size	Page_Title			Server_Type F
http://www.ibm.com	785	13071	13	662003	IBM - United Stat	es		IBM_HTTP_Server 2
http://www.watson.ibm.com	89	12580	9	5592	IBM Research   Redirect			IBM_HTTP_Server/7.0.0.21 (Unix)
http://www.eclipse.org	656	20598	19	266444	Eclipse - The Eclip	se Foundation open source	e community website.	Apache 2
41								
] •								

#### 图 38. "代理程序测试"透视图

如果未显示数据,则"**属性组测试**"视图中会显示以下消息:返回了 0 个数据行。导致代理程序不返回数据的原因有多种。这些原因包括:

·没有数据

·定义不正确

·配置不正确

您可以通过查看 Performance Object Status 属性组中的 Error\_Code 值来检查未返回数据的原因。有 关查看 Performance Object Status 属性组的更多信息,请参阅步骤<u>第 197 页的『9』</u>。

要收集运行中的代理程序中另一属性组的数据,请选择所需的属性组。

在"代理程序测试"视图中选择属性组时,对应的属性组会显示在 Agent Editor 视图中。

5. 可选: 对特定属性组类型运行首次数据收集之后,再次运行数据收集以获取有用的数据值。

要运行数据收集,请单击"属性组测试"视图中的"收集数据"图标 💞 。

如果收集需要一些时间,那么窗口将指示数据收集正在进行中。数据收集完成时,新收集的数据显示在 "**属性组测试**"视图中。

6. 可选: 单击 "属性组测试"视图中的特性列标题以打开 Agent Editor 的"数据源定义"选项卡中的 "特性信息"。您也可以通过右键单击表中的任意数据单元格,并从菜单中选择编辑来访问同一个"特性信息"。

您可以按常规方式编辑特性的属性。直到运行中的代理程序停止并重新启动后,您所做的更改才会显示在该代理程序中。

7. 可选: 同时打开多个"属性组测试"视图。

要再打开一个"**属性组测试"**视图,请单击"**属性组测试"**视图工具栏上的"视图菜单"图标 🎽,然 后选择**打开属性组的视图**。

**注:** 另外打开一个"**属性组测试"**视图时,它将显示与原始"**属性组测试"**视图相同的特性信息。然后,您可以在"**代理程序测试"**视图中选择另一个属性组,以在原始"**属性组测试"**视图中显示不同的属性组信息。首次打开另一个"**属性组测试"**视图时,它会在与原始视图相同的位置处打开,但具有其自己的选项卡。如果您希望同时查看这两个视图,可以将该选项卡拖动到工作空间中的另一位置。

8. 可选: 选择子节点实例信息属性组(如果代理程序具有子节点),以查看子节点在代理程序中的列示方式(<u>第195页的图37</u>)。选择子节点实例信息属性组会在"**属性组测试**"视图中显示子节点实例信息 (针对所选类型的所有联机子节点)。 9. 可选: 要查看有关代理程序运行的更多信息, 可以在"代理程序测试"视图中选择 Performance

Object Status 和 Thread Pool Status 属性组。这些状态属性组通过"信息"图标 i 指示。选择这些组 可以查看有关属性组的先前数据收集的状态信息。

1-1-1	[ I	
<i>kh</i>	$H \Pi$	•
TAI	UU	
17.1	거머	•

12 3 2 4 1 -											
Attribute Group Test: X										🤣 🗸 🗖 🗖	
Data collection at 10-Sep-2012 14:23:52 returned 3 data rows.											
Query_Name	Object_Name	Object_Type	Object_Status	Error_Code	Last_Collection_Start	Last_Collection_Finished	Last_Collection_Duration	Average_Collection_Duration	Refresh_Interval	Number_of_Collections	Cache_Hits
URL_Objects	URL_Objects	CUSTOM	ACTIVE	NO_ERROR	10-Sep-2012 14:23:21	10-Sep-2012 14:23:42	20.67	20.67	0	1	0
Managed_URLs	Managed_URLs	CUSTOM	ACTIVE	NO_ERROR	10-Sep-2012 14:23:00	10-Sep-2012 14:23:14	13.33	16.84	0	4	0
•											•

图 39. "属性组测试"视图,显示与 Managed\_URLs 和 Managed\_Nodes 属性组的数据收集有关的更 多信息 (Performance Object Status)。

10. 在完成代理程序测试时,单击"停止代理程序"图标 📕。

# 测试环境变量

使用以下环境变量可以控制代理程序在测试期间的行为。

环境变量是动态的具名值,用于确定代理程序的运行方式。对于属性组测试,部分代理程序环境变量设置为 特殊值。使用特殊值的目的,是让代理程序以适合测试单个属性组的方式进行响应。对于代理程序全面测 试,不使用特殊值,而是使用缺省值。缺省值意味着代理程序的行为与平常一样,这样更适用于代理程序全 面测试。

下表汇总具有用于属性组测试的特殊值的环境变量。有关所有代理程序环境变量的更多信息,请参阅第19 页的『环境变量列表』。有关设置环境变量的更多信息,请参阅第19页的『环境变量』。

表 43. 环境变量			
环境变量	缺省值(代 理程序全面 测试)	属性组测试值	更改属性组测试值的原因
CDP_DP_INITIAL_COLLECTI ON_ DELAY	不固定	1	此值适用于带有线程池的代理程序。此值是将 初始数据收集请求发送到数据提供程序之前, 线程池等待的时间(秒)。
			<b>注:</b> 如果未设置 CDP_DP_INITIAL_COLLECTION_DELAY,那 么线程池将等待 CDP_DP_REFRESH_INTERVAL或 CDP_ATTRIBUTE_GROUP_REFRESH_INTERV AL所指定的时间。此等待时间就是线程池在两 次数据收集之间等待的时间,对于等待第一次 数据收集来说可能太长。
CDP_DP_CACHE_TTL	55	1	设置为1时, "收集数据"请求极有可能导致 数据提供程序立即收集数据。否则, 它可能会 返回过去最多 60 秒的高速缓存数据。

# 第14章 在监视基础结构中安装代理程序以供测试和使用

在 Agent Builder 中测试代理程序之后,可以将代理程序安装到现有的 IBM Tivoli Monitoring 或 IBM Cloud Application Performance Management 环境中,以供进一步测试和使用。

在监视基础结构中安装和测试代理程序有下列优势:

- ·可以配置和测试同时运行的多个代理程序实例。
- ·可以配置和测试同时运行的多个子节点实例。
- ·在 Tivoli Monitoring 环境中,可以在 Tivoli Enterprise Portal 中构建工作空间、情境、操作和查询。

**要点:**将代理程序的初始版本部署到监视基础结构的测试版本中。在 Tivoli Monitoring 上,请使用单独的监视服务器和门户网站服务器。在 Cloud APM 上,请使用测试云帐户,或使用本地监视服务器的单独测试部署。将代理程序的最终版本部署在生产基础结构上。

如果将代理程序的某个版本部署在生产监视基础结构上,然后更改该代理程序中的任何数据集,那么新版本可能会与服务器上的旧版本发生冲突。在这种情况下,可能无法使用该代理程序的任何版本。

# 安装代理程序

可以使用两种方法来安装使用 Agent Builder 创建的代理程序。

- 1. 要使用与 Agent Builder 在同一系统上运行的监视基础结构来测试代理程序,可以将代理程序安装到本地 Tivoli Monitoring 或 Cloud APM 安装。
- 2. 要使用不与 Agent Builder 在同一系统上运行的 Tivoli Monitoring 或 Cloud APM 系统来测试或使用代理程 序,可以生成能够传输到其他系统的压缩文件(代理程序包)并进行部署。

注:

- 1. 对于 Tivoli Monitoring,在安装代理程序之后,可以在 Tivoli Enterprise Portal 表中查看性能指标。有关 情境或工作空间的支持,请参阅<u>第 217 页的『第 15 章 导入应用程序支持文件』</u>。
- 2. 对于 Tivoli Monitoring,在安装代理程序之后,可以使用 Tivoli Enterprise Portal 验证来自代理程序的数据。有关更多信息,请参阅<u>第 209 页的『Tivoli Enterprise Portal 中的更改』</u>。如果在 Tivoli Enterprise Portal 中查看数据之后要修改代理程序,请参阅第 17 页的『第 4 章 使用 Agent Editor 修改代理程序』。
- 3. 对于支持 Linux 或 UNIX 的代理程序,请在 Linux 或 UNIX 系统上生成安装程序映像,因为 Linux 或 UNIX 系统会创建具有相应许可权的文件。

# 本地安装代理程序

将代理程序安装到运行 Agent Builder 的本地系统上的监视环境中。

# 关于此任务

完成以下步骤以将代理程序安装到本地系统上的监视环境中:

- 1. 使用下列其中一种方法, 在 Agent Builder 的 Project Explorer 导航树中单击 itm\_toolkit\_agent.xml 文件:
  - a. 右键单击 itm\_toolkit\_agent.xml 文件并选择 IBM > 生成代理程序。
  - b. 选择 itm\_toolkit\_agent.xml 文件,并选择工具栏上的 📚 生成代理程序图标。
  - c. 双击 itm\_toolkit\_agent.xml 文件并选择 Agent Editor > 生成代理程序。
- 2. 在"**生成代理程序向导**"窗口的**本地安装代理程序**部分中,输入监视基础结构的安装目录。Agent Builder 填写 CANDLE\_HOME 环境变量中找到的值。如果未设置此变量,那么将显示 Windows 的缺省值 C:\IBM\ITM。

各个复选框按如下方式启用:

安装代理程序

如果 Agent Builder 在指定位置中检测到相应的 Tivoli Enterprise Monitoring Agent 或 IBM Cloud APM 代理程序,那么启用。相应的代理程序是指支持本地操作系统并且处于正确的最低版本的代理程序。

安装 TEMS 支持

如果 Agent Builder 在指定位置中检测到 Tivoli Enterprise Monitoring Server, 那么在 Tivoli Monitoring 环境中启用。

安装 TEPS 支持

如果 Agent Builder 在指定位置中检测到 Tivoli Enterprise Portal Server, 那么在 Tivoli Monitoring 环 境中启用。

- 3. 选择要安装的组件(代理程序、Tivoli Enterprise Monitoring Server 支持和 Tivoli Enterprise Portal Server 支持)。
- 4. 在 Tivoli Monitoring 环境中,如果 Tivoli Enterprise Monitoring Server 或 Tivoli Enterprise Portal Server 安装在本地计算机上,并且您要安装这些服务器的支持文件,那么可以选择是否重新启动服务器。

在此情况下, "生成代理程序向导"的本地安装代理程序部分中的重新启动 TEMS 而无凭证和重新启动 TEPS 复选框处于活动状态。您可以将这些复选框取消选中,以便在不重新启动服务器的情况下安装此支持。

在清除**重新启动 TEMS 而无凭证**复选框时,将提示输入 Tivoli Enterprise Monitoring Server 用户标识和 密码。输入这些详细信息,然后单击**登录**。如果要在关闭安全性的情况下运行 Tivoli Monitoring,请输入 "sysadmin" 作为用户标识,密码留空,然后单击**登录**。

或者,您也可以继续操作而不输入凭证,不指定用户标识和密码并单击登录,或者单击取消。如果完成这些步骤,那么 Tivoli Enterprise Monitoring Server 将重新启动。

**要点:** 要在不重新启动 Tivoli Enterprise Monitoring Server 的情况下安装支持文件,请确保 Tivoli Enterprise Monitoring Server 处于正在运行状态。

- 5. 选择要生成的代理程序组件。您可以选择基本代理程序和/或 Cognos Reporting。
- 6. 在 IBM Cloud APM 环境中,您可以为自描述代理程序提供安全签名。单击**编辑所有 JAR 签名首选项**。您可以将时间戳记添加到签名 JAR 文件并指定添加时间戳记权限。指定有关 Java 密钥库文件的详细信息。

注: 您必须使用 Java 工具来创建 Java 密钥库文件。例如,要使用 Java 密钥库文件中的对应公用密钥来 生成专用密钥和证书,可以运行以下命令:

·ab\_install\_path/jre/bin/keytool -genkeypair -keystore keystore\_file\_path storepass key\_store\_password -alias key\_store\_alias -dname "CN=common\_name, OU=organizational\_unit, L=city\_or\_locality, ST=state\_or\_province, C=country" -keypass key\_password

其中:

·ab\_install\_path 是 Agent Builder 的安装位置

·keystore\_file\_path 是现有 JKS 密钥存储库的路径,或者是要在其中创建 JKS 密钥存储库的路径

·key\_store\_password 是访问此密钥库中的任何项所需的密码

- ·key\_store\_alias 是用于在密钥库中识别此密钥的名称(缺省为"mykey")
- ·key\_password 是访问此特定密钥所需的密码(缺省为 key\_store\_password)

证书必须包括在服务器的密钥库中。

- 7. 填写完"JAR签名"详细信息时,单击确定。
- 8. 单击完成。
- 9. 配置并启动代理程序。有关更多信息,请参阅 IBM Cloud APM 环境中的<u>第 203 页的『在 IBM Tivoli</u> Monitoring 环境中配置和启动代理程序』或第 204 页的『配置代理程序』和第 205 页的『启动和停止代 理程序』。

对于 Tivoli Monitoring V6.2 FP1 或更高版本,您无需重新启动服务器即可安装 Tivoli Enterprise Monitoring Server 和 Tivoli Enterprise Portal Server 支持。在此情况下, "生成代理程序向导"的本地安 装代理程序部分中的重新启动 TEMS 而无凭证和重新启动 TEPS 复选框处于活动状态。您可以将这些复
选框取消选中,以便在不重新启动服务器的情况下安装此支持。在清除**重新启动 TEMS 而无凭证**复选框时,将提示您输入 Tivoli Enterprise Monitoring Server 用户标识和密码。请输入 Tivoli Enterprise Monitoring Server 用户标识和密码并单击**登录**。如果要在关闭安全性的情况下运行 Tivoli Monitoring,请输入 "sysadmin"作为用户标识,密码留空,然后单击**登录**。您也可以继续操作而不输入凭证(不指定用户标识和密码并单击**登录**,或者单击**取消**。这样将导致重新启动 Tivoli Enterprise Monitoring Server)。

**注:** 要在不重新启动 Tivoli Enterprise Monitoring Server 的情况下安装支持文件, Tivoli Enterprise Monitoring Server 必须处于运行状态。

# 创建代理程序包

您可以使用 Agent Builder 来创建压缩代理程序安装包。

# 关于此任务

代理程序包中包含运行代理程序所需的所有文件以及安装和配置脚本。此安装包还包含监视环境的支持文件。

您可以使用代理程序包将代理程序安装到 IBM Tivoli Monitoring 和 IBM Cloud Application Performance Management 环境中。

# 过程

- 1. 使用下列其中一种方法, 在 Agent Builder 的 **Project Explorer** 导航树中单击 itm\_toolkit\_agent.xml 文件:
  - · 右键单击 itm\_toolkit\_agent.xml 文件并选择 IBM > 生成代理程序。
  - ·选择itm\_toolkit\_agent.xml文件,并选择工具栏上的 😫 生成代理程序图标。
  - · 双击 itm\_toolkit\_agent.xml 文件并选择 Agent Editor > 生成代理程序。
- 2. 在生成代理程序映像部分中,输入用于存放输出(压缩包或解压缩文件)的目录的名称。
- 3. 选中保留中间文件复选框,以便将生成的解压缩文件与 zip 或 tar 文件分开存放。
- 4. 选中**创建 ZIP 文件**复选框,以便在指定的目录中创建压缩文件。缺省情况下,对于 Windows 系统,此压 缩 zip 文件名为 smai-agent\_name-version.zip。
- 5. 选中**创建 TAR 文件**复选框,以便在指定的目录中创建 tar 文件。缺省情况下,对于 UNIX 和 Linux 系统, 此压缩 tar 文件名为 smai-*agent\_name-version*.tgz。
- 6. 选择要生成的代理程序组件。您可以选择基本代理程序和/或 Cognos Reporting。

**要点:**对于 IBM Cloud Application Performance Management 环境,请勿选择 **Cognos 报告**,这是因为 这些报告目前不受支持,而包括这些报告会增加包的大小。

7. 您可以选择性地提供代理程序应用程序文件的安全签名。如果您想要提供安全签名,请选择**签署自我描述支持 JAR**。单击编辑所有 JAR 签名首选项。您可以将时间戳记添加到签名 JAR 文件并指定添加时间戳记权限。指定有关 Java 密钥库文件的详细信息。

要点:您可以使用 Java 工具来创建 Java 密钥库文件。例如,要使用 Java 密钥库文件中的对应公用密钥 来生成专用密钥和证书,可以运行以下命令:

```
• ab_install_path/jre/bin/keytool -genkeypair -keystore keystore_file_path -
storepass key_store_password -alias key_store_alias -dname "CN=common_name,
OU=organizational_unit, L=city_or_locality, ST=state_or_province,
C=country" -keypass key_password
```

其中:

- ·ab\_install\_path 是 Agent Builder 的安装位置
- ·keystore\_file\_path 是现有 JKS 密钥存储库所在的路径,或者是要在其中创建 JKS 密钥存储库的路径
- ·key\_store\_password 是访问此密钥库中任何项所需的密码
- ·key\_store\_alias 是用于在密钥库中唯一标识此密钥的名称(缺省为"mykey")

·key\_password 是访问此特定密钥所需的密码(缺省为 key\_store\_password)

请将此证书包括在服务器密钥存储库中。

8. 单击完成。

# 在 IBM Tivoli Monitoring 环境中安装软件包

要在 IBM Tivoli Monitoring 环境中测试和使用代理程序,请使用生成的软件包在受监视的系统、中心 Monitoring Server 系统和 Portal Server 系统上安装代理程序。

### 开始之前

在受监视的系统上安装代理程序之前,请确保 Tivoli Monitoring 操作系统代理程序已存在并且正常工作。有 关安装 Tivoli Monitoring 代理程序的信息,请参阅 Tivoli Monitoring Knowledge Center 中的<u>安装监视代理程</u> <u>序</u>。

要点: 要在 Tivoli Enterprise Portal 中显示代理程序信息,必须安装以下组件:

·将代理程序安装在所有受监视的系统上

- ·将 Tivoli Enterprise Monitoring Server 支持文件安装在中心 Tivoli Enterprise Monitoring Server 上
- ·将 Tivoli Enterprise Portal Server 支持文件安装在 Tivoli Enterprise Portal Server 上
- ·将 Tivoli Enterprise Portal 支持文件安装在 Tivoli Enterprise Portal Server 和(如果适用)任何 Tivoli Enterprise Portal 桌面客户机上

# 过程

- 1. 将缺省情况下名为 product\_code.zip(针对 Windows 系统)或 product\_code.tgz(针对 UNIX 和 Linux 系统)的压缩文件复制到要安装代理程序的系统上。
- 2. 将此文件解压缩至临时位置。

**注:** Linux 对于 UNIX 和 Linux 系统,此临时位置不得为 /tmp/product\_code,其中产 品代码为小写。

可以通过使用压缩文件来远程安装代理程序。

· Linux 在 Linux 系统上,使用以下命令来解压缩.tgz 文件:

tar -xvzf filename

· \_\_\_\_在 AIX 系统上, 使用以下命令来解压缩.tgz 文件:

gunzip filename tar -xvf filename

- 3. 运行相应的安装脚本。
  - · 要同时安装代理程序、Tivoli Enterprise Monitoring Server、Tivoli Enterprise Portal Server和 Tivoli Enterprise Portal 支持文件:

InstallIra.bat/.sh itm\_install\_location [[-h Hub\_TEMS\_hostname] -u
HUB\_TEMS\_username -p Hub\_TEMS\_password]

· 要在不安装支持文件的情况下安装代理程序:

installIraAgent.bat/.sh itm\_install\_location

要安装 Tivoli Enterprise Monitoring Server 支持:

installIraAgentTEMS.bat/.sh itm\_install\_location [[-h Hub\_TEMS\_hostname] -u HUB\_TEMS\_username -p Hub\_TEMS\_password]

· 要安装 Tivoli Enterprise Portal Server 和 Tivoli Enterprise Portal 支持文件:

installIraAgentTEPS.bat/.sh itm\_install\_location

安装位置 *itm\_install\_location* 必须是第一个自变量,并且在所有脚本中均为必需: installIra.bat/.sh、installIraAgent.bat/.sh、installIraAgentTEMS.bat/.sh和 installIraAgentTEPS.bat/.sh。这是此系统上安装 Tivoli Monitoring 组件的位置。

其他自变量为可选。

如果安装 Monitoring Server 支持文件并且不提供用户标识,那么 Tivoli Enterprise Monitoring Server 将 重新启动。

4. 配置并启动代理程序,请参阅第 203 页的『在 IBM Tivoli Monitoring 环境中配置和启动代理程序』。

# 下一步做什么

如果更改了代理程序的布局,导致移动或移除了导航器项,请重新启动 Tivoli Enterprise Portal Server 和 Tivoli Enterprise Portal。重新启动可确保正确识别您所作的更改。

# 在 IBM Tivoli Monitoring 环境中配置和启动代理程序

在 IBM Tivoli Monitoring 中的受监视系统上安装代理程序后,请配置并启动此代理程序。

# 过程

1. 打开管理 Tivoli Monitoring Service。

这将显示新条目 Monitoring Agent for agent\_name。

2. 右键单击该条目并选择使用缺省值进行配置。如果出现提示,请单击确定以接受缺省值。

### 要点:

a. 在 UNIX 系统上,应选择配置选项。

b. 在配置多实例代理程序时,将提示输入实例名称。

提示: 如果您的代理程序使用 JMX 数据源来连接到远程 WebSphere Application Server, 请确保 WebSphere Application Server 也安装在运行该代理程序的主机上,并将 Java home 设置设定为本地 WebSphere Application Server 使用的 Java 运行时环境。

提示: 对于 Java API、JDBC、JMX、HTTP 或 SOAP 数据源,您可以使用 Java > JVM 自变量设置来控制 代理程序跟踪日志记录。在此设置中设置以下值:

-DJAVA\_TRACE\_MAX\_FILES=files -DJAVA\_TRACE\_MAX\_FILE\_SIZE=size

其中, files 是保留的最大跟踪日志文件数量(缺省值为 4), size 是以千字节为单位的最大日志文件大小 (缺省值为 5000)。例如,您可以设置以下值:

-DJAVA\_TRACE\_MAX\_FILES=7 -DJAVA\_TRACE\_MAX\_FILE\_SIZE=100

在此情况下,代理程序将 100 千字节写入第一个日志文件,然后切换到第二个日志文件,以此类推。在写入七个 100 千字节的日志文件之后,将覆盖第一个日志文件。

如果已将运行时配置元素添加到代理程序,或者已选择数据源,那么系统将显示配置面板。您可以使用这些面板来收集代理程序所需的信息。

- 3. 右键单击代理程序条目, 然后选择启动
- 4. 打开 Tivoli Enterprise Portal 并转至新代理程序。

# 在 IBM Cloud Application Performance Management 环境中安装和使用代理程序

要在 IBM Cloud Application Performance Management 环境中测试和使用代理程序,请使用生成的软件包在所有受监视的系统上安装代理程序。在某些情况下,您需要先配置代理程序,然后才能将其启动。您可以根据需要启动和停止代理程序。

### 安装代理程序

使用 Agent Builder 准备的安装程序包在所有受监视系统上安装代理程序。

#### 开始之前

请确保受监视系统上已存在用于 IBM Cloud Application Performance Management 的代理程序(通常是操作系统代理程序),并确保该代理程序正常工作。

Windows 在 Windows 系统上,使用管理员命令行 shell 来安装和配置代理程序。要启动管理员 shell,请从 Windows "程序"菜单选择**命令提示符**并单击右键,然后单击**以管理员身份运行**。

### 过程

1. 将程序包解压缩到临时目录并切换到此目录。

2. 根据操作系统不同,使用以下命令来安装代理程序:

・ Windows 在 Windows 系统上: installIraAgent.bat agent\_install\_location

· Linux 在 Linux 和 UNIX 系统上: ./installIraAgent.sh

agent\_install\_location

其中, agent\_install\_location 是现有代理程序的安装位置。缺省位置为:

· Windows 在 Windows 系统上: C:\IBM\APM

· Linux 在 Linux 系统上: /opt/ibm/apm/agent

· \_ AX 在 AIX 系统上: /opt/ibm/apm/agent

**要点:**如果您已在 Agent Editor 的"运行时配置"窗口中添加了任何定制配置属性,如果代理程序支持多个实例,或者如果代理程序使用任何需要配置(例如,用户标识和密码)的预定义数据源,那么您必须 先配置代理程序,然后它才能启动。如果代理程序不需要配置,那么完成安装之后,它将自动启动。

#### 配置代理程序

如果您已在 Agent Editor 的"运行时配置"窗口中添加了任何定制配置属性,如果代理程序支持多个实例, 或者如果代理程序使用任何需要配置(例如,用户标识和密码)的预定义数据源,那么您必须先配置代理程 序,然后它才能启动。

### 开始之前

Windows 在 Windows 系统上,使用管理员命令行 shell 来安装和配置代理程序。要启动管理员 shell,请从 Windows "程序"菜单选择命令提示符并单击右键,然后单击以管理员身份运行。

### 关于此任务

在配置过程中,您可以执行以下操作:

- ·如果代理程序支持多个实例,可以设置实例名称以创建或更改实例。
- ·设置任何对代理程序可用的配置属性。
- ·如果代理程序支持子节点,创建和配置子节点。

Windows 在 Windows 系统上,要设置任何配置属性或创建任何子节点,必须使用静默配置过程。样本静默 配置响应文件位于 *install\_dir*\samples 目录中,名为 *agentname\_*silent\_config.txt。创建此文 件副本,并根据需要设置配置变量。

**Linux**在 Linux和 UNIX系统上,您可以选择使用静默配置过程。或者,可以使用交互式过程。如果不使用响应文件名称来启动配置命令,配置实用程序将提示输入配置值。

# 过程

- 1. 切换到 install\_dir/bin 目录。
- 2. 运行以下命令以配置代理程序:
  - · 如果代理程序不支持多个实例:
    - Windows 在 Windows 系统上: name-agent.bat config [response\_file]
    - Linux UNIX 在 Linux 和 UNIX 系统上: ./name-agent.sh config [response\_file]
  - · 如果代理程序支持多个实例:
    - Windows 在 Windows 系统上: name-agent.bat config instance\_name [response\_file]
    - Linux 在 Linux和 UNIX系统上: ./name-agent.sh config instance\_name [response\_file]

其中:

·*instance\_name* 是实例名称。如果不存在具有此名称的实例,将创建此实例。如果已存在此实例,将重新配置此实例。您必须至少创建一个实例来使用代理程序。

·response\_file 是静默配置响应文件的名称。

提示: 如果您的代理程序使用 JMX 数据源来连接到远程 WebSphere Application Server, 请确保 WebSphere Application Server 也安装在运行该代理程序的主机上,并将 Java home 设置设定为本地 WebSphere Application Server 使用的 Java 运行时环境。

提示: 对于 Java API、JDBC、JMX、HTTP 或 SOAP 数据源,您可以使用 Java > JVM 自变量设置来控制 代理程序跟踪日志记录。在此设置中设置以下值:

-DJAVA\_TRACE\_MAX\_FILES=files -DJAVA\_TRACE\_MAX\_FILE\_SIZE=size

其中, files 是保留的最大跟踪日志文件数量(缺省值为 4), size 是以千字节为单位的最大日志文件大小 (缺省值为 5000)。例如,您可以设置以下值:

-DJAVA\_TRACE\_MAX\_FILES=7 -DJAVA\_TRACE\_MAX\_FILE\_SIZE=100

在此情况下,代理程序将100千字节写入第一个日志文件,然后切换到第二个日志文件,以此类推。在写入七个100千字节的日志文件之后,将覆盖第一个日志文件。

# 启动和停止代理程序

要监控系统,请确保在系统上已启动此代理程序。您可以随时启动和停止代理程序。如果代理程序支持多个实例,可以单独启动和停止每个实例。

# 过程

- 1. 切换到 install\_dir/bin 目录。
- 2. 运行以下命令以启动代理程序:
  - · 如果代理程序不支持多个实例:
    - Windows 在 Windows 系统上: name-agent.bat start
    - Linux 在 Linux和 UNIX系统上: ./name-agent.sh start
  - · 如果代理程序支持多个实例:
    - Windows 在 Windows 系统上: name-agent.bat start instance\_name
- Linux 在 Linux 和 UNIX 系统上: ./name-agent.sh start instance\_name
   3.运行以下命令以停止代理程序:
  - · 如果代理程序不支持多个实例:

- Windows 在 Windows 系统上: name-agent.bat stop

- Linux 在 Linux和 UNIX系统上: ./name-agent.sh stop

· 如果代理程序支持多个实例:

- Windows 在 Windows 系统上: name-agent.bat stop instance\_name
- Linux 在 Linux和 UNIX系统上: ./name-agent.sh stop instance\_name

# 生成及安装代理程序之后的结果

安装 Agent Builder 代理程序会创建和更改系统上的某些文件。在 IBM Tivoli Monitoring 环境中,您还会看到 Tivoli Enterprise Portal 中的更改。

### 系统上的新文件

生成并安装使用 Agent Builder 创建的代理程序之后,您在代理程序系统上可以看到下列新文件:

注: xx 表示两个字符的产品代码。

#### Windows

Windows系统: TMAITM6\kxxagent.exe

代理程序二进制文件

TMAITM6\KxxENV 环境变量设置

**TMAITM6\K***xx*.ref 代理程序提供程序配置

**TMAITM6\SQLLIB\k***xx*.his 代理程序属性信息的 SQL 描述

**TMAITM6\SQLLIB\k***xx*.atr 代理程序属性信息

TMAITM6\*xx*\_dd\_*version*.xmll 产品描述

TMAITM6\xx\_dd.properties

产品名称

TMAITM6\kxxcma.ini

代理程序服务定义文件

### TMAITM6\your files

通过 Java API 或套接字数据源附带的补充文件,其文件类型为"可执行文件"或"库"。通过脚本或命 令返回码数据源包括的脚本。

Linux UNIX

### UNIX/Linux 系统: registry/xxarchitecture.ver

内部版本和必备文件

architecture/xx/bin/xx\_dd\_version.xml

产品描述

*architecture/xx/bin/kxx*agent 代理程序二进制文件

architecture/xx/bin/xx\_dd.properties 产品名称 architecture/xx/work/kxx.ref 代理程序提供程序配置 architecture/xx/tables/ATTRLIB/kxx.atr 代理程序属性信息 architecture/xx/hist/kxx.his 代理程序属性信息的 SQL 描述 architecture/xx/bin/your files 通过 Java API 或套接字数据源附带的补充文件,其文件类型为"可执行文件"。通过脚本或命令返回码 数据源包括的脚本。 architecture/xx/lib/your files 通过 Java API 或套接字数据源附带的补充文件,其文件类型为"库"。 config/.xx.rc 内部设置文件 config/xx.environment 环境设置 config/xx\_dd\_version.xml 产品描述 config/xx\_dd.properties 产品名称 config/.ConfigData/kxxenv 环境变量设置 注:请运行以下命令,以确定系统的体系结构: cinfo -pxx 其中 xx 是两个字符的产品代码。

例如,对于以产品代码 19 来运行代理程序 的 64 位 Solaris 8 系统,输出如下所示:

# /opt/ibm/apm/agent/bin/cinfo -p 19

以粗体显示的行是相关行。冒号之前的字符串 sol286 指示用于此代理程序的体系结构。对于操作系统与计算机硬件的不同组合,此字符串有所不同。只有在先前已安装代理程序的情况下,此功能才有效。

下列文件用于基于 Java 的数据源。只有在代理程序包含 JMX、JDBC、HTTP 或 SOAP 数据源的情况下,才 会创建这些文件:

•cpci.jar

·jlog.jar

·common/jatlib-1.0.jar

下列文件用于 JMX 运行时支持。只有在代理程序包含 JMX 数据源的情况下,才会创建这些文件:

• common/jmx-1.0.jar

・common/connectors/jboss/connJboss-1.0.jar

・common/connectors/jsr160/connJSR160-1.0.jar

・common/connectors/was/connWas-1.0.jar

・common/connectors/weblogic/connWeblogic-1.0.jar

下列文件用于 JDBC 运行时支持。只有在代理程序包含 JDBC 数据源的情况下,才会创建这些文件:

・common/jdbc-1.0.jar

下列文件用于 HTTP 或 SOAP 运行时支持。只有在代理程序包含 HTTP 或 SOAP 数据源的情况下,才会创建 这些文件:

∙http-1.0.jar

下列文件用于 Java API 运行时支持。只有在代理程序包含 Java API 数据源的情况下,才会创建这些文件: ·cpci.jar

·custom/your JAR file:此 JAR 文件的名称在 Java API 数据源的全局设置中指定。

·custom/your JAR file: 文件类型为 Java 资源的补充文件。

对于基于 Java 的数据源,在 Windows、UNIX 和 Linux 系统上存在相同的文件,但它们位于不同的目录中:

· Windows Windows 路径: TMAITM6\kxx\jars

· Linux UNIX UNIX/Linux 路径: architecture/xx/jars

下列文件用于日志文件监视运行时支持。只有在代理程序包含日志文件数据源的情况下,才会创建这些文件:

- · Windows 在 Windows 系统上: TMAITM6\kxxudp.dll
- · Linux 在 Solaris/Linux 系统上: architecture/xx/lib/libkxxudp.so
- ·在 HP-UX 系统上: architecture/xx/lib/libkxxudp.sl
- · \_ uxx 在 AIX 系统上: architecture/xx/lib/libkxxudp.a

下列文件用于 SSH 脚本监视运行时支持。只有在代理程序包含已启用 SSH 收集的脚本数据源的情况下,才 会创建这些文件:

- · Windows 在 Windows 系统上: TMAITM6\kxxssh.dll
- · Linux 在 Solaris/Linux 系统上: architecture/xx/lib/libkxxssh.so
- ·在 HP-UX 系统上: architecture/xx/lib/libkxxssh.sl
- · **\_\_\_\_**在 AIX 系统上: architecture/xx/lib/libkxxssh.a

# 管理 Tivoli Enterprise Monitoring Services 窗口中的更改

在 IBM Tivoli Monitoring 环境中安装代理程序之后,您可以在"**管理 Tivoli Enterprise Monitoring** Services"窗口中看到一个代理程序条目。该条目名称为 Monitoring Agent for *agent\_name*。

**要点: "管理 Tivoli Enterprise Monitoring Services**"在 IBM Cloud Application Performance Management 环境中不受支持。

Windows 在 Windows 系统上,此条目包含任务/子系统列,用于标识代理程序是否支持多个实例:

- ·单实例代理程序将在"管理 Tivoli Enterprise Monitoring Services"窗口中显示新的应用程序。此应用程 序名称为 Monitoring Agent for *agent\_name*。针对此代理程序,将创建一个服务(<u>第 209 页的图 40</u>)。 任务/子系统列包含值主要。
- ·多实例代理程序将在"管理 Tivoli Enterprise Monitoring Services"窗口中显示新的应用程序模板。此模板名称为 Monitoring Agent for agent\_name。根据此模板创建此代理程序的实例之后,才会为此代理程序创建服务。任务/子系统列包含值模板,以指示此条目是用于创建代理程序实例的模板。

Linux LINUX 在 Linux 和 UNIX 系统上,无论代理程序是否支持多个实例,代理程序的条目都相同。

注: 下列屏幕用于 Windows 系统。在 UNIX 和 Linux 系统上,存在类似的屏幕。

2										
Security Configurati.										
	Manage Tivoli Enterprise Monitoring Services - TEMS Mode - [Local Computer]									
	Actions Options View Windows Help									
Recycle Bir	III III 🔁 💆 🖉									
	Service/Application	Task/SubSystem	Configured	Status	Startup	Account	Desktop	HotStdby	Version	
	🕏 🖻 Eclipse Help Server	HELPSVR	Yes	Stopped	Auto	LocalSystem	No	No	3.0.1	
	Tivoli Enterprise Portal	Browser	Yes		N/A	N/A	N/A	N/A	06.20.00	
Tivoli	🔛 Tivoli Enterprise Portal	Desktop	Yes		N/A	N/A	N/A	N/A	06.20.00	
Enterpr	Tivoli Enterprise Portal Server	KFWSRV	Yes (TEMS)	Started	Auto	LocalSystem	No	No	06.20.00	
	Universal Agent	Primary	Yes (TEMS)	Started	Auto	LocalSystem	No	No	06.20.00	
	Monitoring Agent for Windows OS	Primary	Yes (TEM5)	Started	Auto	LocalSystem	Yes	No	06.20.00	
	Tiveli Enterprice Manitoring Server	TEMS1	Yes (TEMD)	Started	Auto	LocalSystem	No	No	06.20.00	
	1								Þ	
	<u></u>									
A Start	🔞 🙆 🗏 Manage Tiyoli Enteror									0
Jocart	🕲 🐷 🗍 📓 Manage Hyon Encerpr									

图 40. "管理 Tivoli Enterprise Monitoring Services"窗口

# Tivoli Enterprise Portal 中的更改

在 IBM Tivoli Monitoring 环境中,安装并启动代理程序后,单击 Tivoli Enterprise Portal 中的绿色刷新图标。然后,您就可以查看新的代理程序。您在门户网站中可以看到以下更改:

- · Tivoli Enterprise Portal 物理视图中新的代理程序子节点。
- ·您使用 Agent Builder 定义的每个导航器组及数据源的对应节点(<u>第 210 页的图 41</u>)。

注:对于每个导航器项,您必须定义一个缺省查询。

Win32 ShareToDirectory - TKWIN2K3 - SYSADMIN					- 🗆 🗙		
File Ealt View Help			en a an a'				
	0 0 2 4	4 🔳 😏 🗔		1 🛄 🔟 🖳 🖓 🖓 🖅 🛄 💽 🔥 📼			
📲 Navigator 🌲 🗉 🖻	💽 View not def	ined		± 1			
🕘 🦑 View: Physical 🔽	🕈 🔶 🔘 👶	🖁 🚰 📇 Ka Locat	on: win2k3	3:1920///cnp/kdh/lib/classes/candle/fw/resources/help/view	_notdefined.htm		
	View not de The default wor a URL in the ac described in the Hands-on prac Tutorial: I Using work Customizin	fined kspace for this Navig Idress text box to op ese topics: tice and overviews Defining a workspace spaces g workspaces	jator item en a We	m contains this <i>browser vie</i> w and a <i>table view</i> . You can enter eb page. You can also change to a different view, as View Choices Involi Enterprise Console event viewer Table view S III I I I I I I I I I I I I I I I I I			
Physical	Done						
Report				/ ¥			
Node Share		Timestamp	1.7.1.0.0.00.0	SharedElement			
TKWIN2K3:55 UTKWIN2K3:roottcimv2:Win32_Share.http://www.stanare.http:/	vame="C\$" Jame="ADMIN\$"	07/10/07 17:20:30	UTKWIN:	2K3\root\CIMV2:Win32_Directory.Name="c:\\" 2K3\root\CIMV2:Win32_Directory.Name="c:\\windows"			
🕒 Hub Time: Tue, 07/10/2007 05:20	PM 🛛 😳 Se	rver Available		Win32 ShareToDirectory - TKWIN2K3 - SYSADMIN			

图 41. 新代理程序中属性组的节点。

·如果代理程序包含子节点,那么对于代理程序中定义的每个子节点,都表现为一个可展开的节点。在可展开的节点之下,将显示下列节点:

- xxx 性能对象状态,其中 xxx 是以3个字母表示的子节点类型
- 您在子节点中定义的每个导航器组和数据源的对应节点
- xxx 事件日志节点(如果有事件日志)
- xxx JMX 监视器节点(如果存在 JMX 且包括了 JMX 监视器)

·下列自动节点:

- 可用性节点(如果代理程序包含可用性数据源)(<u>第 211 页的图 42</u>)

**注:** 根据代理程序的内容不同,此节点的行为也会有所变化。如果代理程序仅监视可用性,那么可用性 节点表示可用性数据源。如果代理程序监视可用性及性能,则可用性节点会变为导航器项,该项代表可 用性及性能对象状态数据源。

Availability - TKWIN2K3 - SY5ADMIN												
	<b>1</b>											
		1 🥸 M 🖸			4	III 🧐 🛄		🕅 🖓 💆 🖸 🖪 🖉 🖌	<b>b C</b>			
📲 Navigator		± 🗉 🖯	P P	erformance	Object S	Status		1	, * 🗆 🖯 🗙			
🔊 🦑 🛛 Vie	w: Physical	*	100	Node	Tim	estamp	Query Name	www.comment.com/Object.Name	Object			
Enterprise		<b></b>	TKW	IN2K3:55	07/10/0	07 17:21:36	Win32_ShareToDirecto	ry ROOT\CIMV2:Win32_Share	eToDirectory WMI			
Windows Systems			TKM	INZK3.55	07710/0	J/ 17.21.30	Browser	Browser	PERFI			
Agent Buil	der		120									
- D Availa	bility		- 12:2									
Brows	er L		1100									
Win32	Log ShareToDirect	tory	14									
🖃 😓 My Applica	ation		- 120									
Event	Log		1.12									
Win32	mance ubject a ? LogicalDisk	latus	1.									
😥 🕀 😥 Universal /	Agent	-	12									
	ne 		14									
egg Physical				مربو والمرد والمعه	· · · · · · · ·		and the second second second	and the second second second	Þ			
🛄 Availability									* # 0 8 0 ×			
Node Tir	nestamp	Application Con	nponent	Nam	e <i>r je s</i> e s	Status		ull Name	Type			
TKWIN2K3:55 07/10/	07 17:21:36	Agent Builder		agentbuilde	er.exe	UP	C:\Program Files\IBM\ITM	AgentBuilder\agentbuilder.exe	PROCESS			
TKWIN2K3:55 07/10/	07.17:21:36	Computer Brow	ser	Browser functionstation	ot		C:WVINDOWS\System32\ N/A	svchost.exe	SERVICE EUNCTIONALITY TE			
1.0710	07 17.21.30	System Status	e e se pro	Tunc_test.b	al	TAILED	1975	FONCHONKENTTE				
CHARLES AND												
C. C												
and the state of the												
all for the factor												
Constant States												
Carl Carlos												
and the state of the												
the first first												
	ور بر می در این می ور ا	and a start of the s		1.1.1.1.1.1.1.1.1.1.1.1.1.1.1.1.1.1.1.								
	Hub Time: T	ue, 07/10/2007 0	5:21 PM		🔵 Serv	er Available	Av	ailability - TKWIN2K3 - SYSADMI	N			

图 42. "可用性"节点

- 性能对象状态, 前提是代理程序包含性能监视(非可用性)数据源(<u>第 212 页的图 43</u>)



图 43. "性能对象状态"节点

- 事件日志, 前提是代理程序包含数据源生成日志数据(第 213 页的图 44)



# 图 44. "事件日志"节点

请参阅<u>第 235 页的『附录 C 属性参考』</u>,以获取 Agent Builder 的属性组及属性的相关描述。

# 卸载代理程序

您可以从受监视的主机移除 Agent Builder 生成的代理程序。

# 关于此任务

卸载过程只会卸载代理程序系统中的代理程序。此过程不会卸载任何其他代理程序或任何监视基础结构。

在 IBM Tivoli Monitoring 环境中,您可以使用以下一个过程来移除 Agent Builder 生成的代理程序:

· 第 213 页的『使用 Tivoli Enterprise Portal 移除 Tivoli Monitoring 代理程序』

· 第 214 页的『在不使用 Tivoli Enterprise Portal 的情况下移除 Tivoli Monitoring 代理程序』

在使用上述任意过程移除代理程序后,使用以下过程从 Tivoli Enterprise Portal 中清除该代理程序: <u>第 214</u>页的『从 Tivoli Enterprise Portal 清除 Tivoli Monitoring 代理程序』。

在 IBM Cloud Application Performance Management 环境中,使用以下过程:<u>第 214 页的『卸载 IBM</u> <u>Cloud Application Performance Management 代理程序』</u>。

# 使用 Tivoli Enterprise Portal 移除 Tivoli Monitoring 代理程序

在 IBM Tivoli Monitoring 环境中,您可以使用 Tivoli Enterprise Portal 移除代理程序。

# 开始之前

操作系统代理程序必须正在运行,才能移除所创建的代理程序。

# 过程

要使用 Tivoli Enterprise Portal 移除代理程序,请完成下列步骤:

· 在 Tivoli Enterprise Portal 导航树中,右键单击代理程序,然后选择移除。

# 在不使用 Tivoli Enterprise Portal 的情况下移除 Tivoli Monitoring 代理程序

如果 Tivoli Enterprise Portal 在 IBM Tivoli Monitoring 环境中不可用,那么可以使用操作系统脚本和命令来移除代理程序。

# 过程

要在不使用 Tivoli Enterprise Portal 的情况下从目标系统移除 Agent Builder 生成的代理程序,可以完成以下 任意步骤:

· Windows 在 Windows 系统上,使用以下命令:

cd ITM\_INSTALL/TMAITM6 kxx\_uninstall.vbs ITM\_INSTALL

其中 xx 是代理程序的产品代码

Windows

或者, 在 Windows 系统上,可以使用 cscript.exe 命令来运行卸载脚本。此命令是 vbs 脚本的命令行 界面解析器,不显示窗口;消息将显示在控制台上:

cd ITM\_INSTALL/TMAITM6 cscript.exe kxx\_uninstall.vbs ITM\_INSTALL

Linux UNIX

在 Linux 或 UNIX 系统上,使用在 ITM\_INSTALL/bin 中找到的 uninstall.sh 文件:

uninstall.sh [-f] [-i] [-h ITM\_INSTALL] [product platformCode]

# 从 Tivoli Enterprise Portal 清除 Tivoli Monitoring 代理程序

在 IBM Tivoli Monitoring 环境中,移除代理程序后,来自代理程序的信息的空字段可能保留在 Tivoli Enterprise Portal 中。要移除这些字段,请从 TivoliEnterprise Portal 清除该代理程序。

# 过程

- 1. 确保 Tivoli Enterprise Monitoring Server 和 Tivoli Enterprise Portal Server 已启动并正在运行。
- 2. 登录到 Tivoli Enterprise Portal 客户机。
- 3. 从 Tivoli Enterprise Portal 客户机的"物理导航器"视图中,右键单击企业,然后选择工作空间 > 受管系 统状态。

此时将显示"受管系统状态"工作空间。

- 4. 选择代理程序的所有 IBM Tivoli Managed System。
- 5. 单击右键并选择清除脱机条目,这样会从表中清除所有上述条目。

# 卸载 IBM Cloud Application Performance Management 代理程序

您可以从 IBM Cloud Application Performance Management 环境中的任何受监视系统卸载自己的代理程序。

过程

- 1. 在安装了代理程序的系统上, 启动命令行并切换到 install\_dir/bin 目录, 其中 install\_dir 是监视代 理程序的安装目录。
- 2. 要卸载特定监视代理程序,请输入代理程序脚本名称和卸载选项,其中 name 是代理程序脚本名称:

·在Windows系统上: *name*-agent.bat uninstall

·在Linux或AIX系统上: ./name-agent.sh uninstall

# 第15章导入应用程序支持文件

如果代理程序要在 IBM Tivoli Monitoring 环境中使用,那么可以在安装程序包中包含定制情境、工作空间、"执行操作"命令和查询。

# 关于此任务

为了使情境、工作空间和代理程序使用一个安装映像,情境文件和工作空间文件必须与代理程序位于同一个项目中。Agent Builder 提供了在代理程序项目中创建适当文件的向导。

还可以在安装程序包中包含与代理程序关联的定义。这些定义的内容对于企业监视环境和系统监视器环境中 使用的代理程序是不同的。企业监视代理程序映像可以包含定制情境、工作空间、"执行操作"命令和查 询。系统监视器代理程序映像可以包含专用情境、陷阱定义和代理程序配置信息。

为将适合的定义和代理程序本身包含在一个安装程序包中,这些文件必须与代理程序位于同一个项目中。 Agent Builder 提供了为企业监视安装创建适当文件的向导。可以使用 *IBM Tivoli Monitoring Administrator's Guide* 中的 *Agent Autonomy* 一章所描述的过程来创建系统监视器代理程序环境的文件。生成的文件将复制 到该代理程序的 Eclipse 项目的根目录。

# 导出和导入 Tivoli Enterprise Monitoring Agent 的文件

# 关于此任务

在 Tivoli Enterprise Portal 中创建情境、工作空间、查询和"执行操作"命令之后,可以将它们一起导出并 导入到 Tivoli Monitoring V6.2 环境中。有关创建情境和工作空间的更多信息,请参阅<u>第 181 页的『第 11 章</u> 创建工作空间、"执行操作"命令和情境』。使用以下步骤可抽取情境、工作空间、"执行操作"命令和查询:

# 过程

- 1. 从 Project Explorer 选项卡,右键单击代理程序项目文件夹。
- 2. 选择 IBM Corporation > 导入应用程序支持文件。
- 3. 输入 Tivoli Enterprise Portal Server 的主机名。
- 4. 输入要连接到的 Tivoli Monitoring 环境的用户名和密码,然后单击完成。
- 5. 如果为代理程序定义了情境,将呈现一个对话框,其中列出了为代理程序定义的情境。
- 6. 从列表中选择要导出的情境,单击 << 以添加到"选定的情境"表,然后单击确定。

导入过程可能需要一段时间。任务完成时,将在代理程序项目中的相应文件夹下显示 SQL 文件。

7. 如果已为代理程序定义了"执行操作"命令,那么将显示一个对话框,其中呈现已定义的"执行操作" 命令。从列表中选择要导出的"执行操作"命令,单击>> 以添加到"选择的执行操作"表,然后单击确 定。

导入过程可能需要一段时间。任务完成时,将在代理程序项目中的相应文件夹下显示 SQL 文件。

8. 如果已为代理程序定义了定制查询,那么将显示一个对话框,其中呈现已定义的"查询"。从列表中选择要导出的查询,单击 << 以添加到"选定的查询"表,然后单击确定。

导入过程可能需要一段时间。任务完成时,将在代理程序项目中的相应文件夹下显示 SQL 文件。工作空间将自动导入。

# 下一步做什么

重新创建定制代理程序,在受监视的主机上安装代理程序,并安装 Tivoli Enterprise Portal 支持。

# 导出和导入 Tivoli System Monitor Agent 的文件

# 关于此任务

系统监视器代理程序定义包含在以下三种类型的文件中:

- ·专用情境在名为 xx\_situations.xml 的文件中定义,其中 xx 是两个字符的产品代码
- · 陷阱配置信息在名为 xx\_trapcnfg.xml 的文件中定义,其中 xx 是两个字符的产品代码
- ·对于需要配置的代理程序,将为代理程序的每个实例在一个文件中定义配置。如果代理程序是单实例代理 程序,那么该文件名为 xx.cfg。如果代理程序是多实例代理程序,那么每个实例都有一个对应文件。文 件名为 xx\_instance name.cfg,其中 xx 是两个字符的产品代码,instance name 是代理程序实例的名称。

### 过程

通过使用 *IBM Tivoli Monitoring Administrator's Guide* 中的 *Agent Autonomy* 一章所描述的过程来创建文件。手动将这些文件复制到项目目录的根目录,或者使用 Eclipse 导入功能来选择要导入的文件: 文件 > 导入 > 常规 > 文件系统。

这些文件包含在代理程序映像中并由安装程序安装。

安装代理程序时,安装过程会:

- 将包含的文件复制到相应的位置。
- 在代理程序上运行的 pc\_situations.xml 文件中定义的任何专用情境。
- pc\_trapcnfg.xml 中定义的陷阱定义用于根据情境转发陷阱。
- 在以下情况下, 将自动配置和启动代理程序:
  - 代理程序是单实例代理程序,并且没有定义为代理程序一部分的配置。
  - 代理程序是单实例代理程序,具有定义为代理程序一部分的配置,且映像包含 pc.cfg 文件。
  - 代理程序是多实例代理程序(所有多实例代理程序都需要配置): 安装程序将为每个 pc\_inst.cfg 文件启动一个代理程序实例。

# 第16章事件过滤和摘要

属性组被定义为纯事件或采样属性组。纯事件属性组包含异步出现的数据行。每个新数据行到达时,Tivoli Monitoring都会立即对其进行处理。采样属性组会在每次请求数据时收集最新的一组数据行。以下属性组说明了它们之间的不同:

- ·创建了 SNMPEvent 属性组以表示发送至代理程序的所有 SNMP 陷阱和通知。陷阱或通知在由受监视系统 发送时异步到达。每个事件到达时,都将传递到 Tivoli Monitoring。
- · 创建了 Disk 属性组以表示系统上所有磁盘的相关信息。磁盘信息是定期收集的。每次收集磁盘信息时, 代理程序都会返回若干数据行(每个磁盘对应一行)。

纯事件属性组和采样属性组之间的不同会影响 Tivoli Monitoring 的各个方面。这些方面包括:情境、仓库数 据和 Tivoli Enterprise Portal 视图。

每个情境指定(或分发)给一个或多个受管系统,以监视一组条件的特定情况。如果必须根据特定时间间隔 的观察结果来确定事件,那么此类事件称为采样事件。如果事件是自然发生的,那么此类事件称为纯事件。 因此,采样事件的情境具有一个关联的时间间隔,而纯事件情境没有。采样事件的另一个特征是导致事件的 条件可以更改,从而使其不再为 true。纯事件无法更改。因此,针对采样事件发起的警报可以从 true 更改为 false,而纯事件在发生时保持为 true。

采样事件的示例是进程数 > 100。当进程数超过 100 时事件变为 true,而稍后当此计数回落至小于等于 100 时将再次变为 false。监视用户的无效登录尝试的情境是一个纯事件;检测到无效登录尝试时该事件发 生,且不会变为 False 事件。虽然可创建根据采样属性组的特定时间间隔进行评估的情境,但此类评估不可 用于纯事件属性组。

类似地,对于历史数据,可以配置收集采样数据的频率。但是,当开启对纯事件数据的收集时,将在每次收 集时获取一行。

Tivoli Enterprise Portal 中针对采样数据显示的数据是最新收集的一组行。针对纯事件属性组显示的数据是 代理程序维护的本地高速缓存的内容。它不一定与传递至 Tivoli Monitoring 以进行情境评估和历史数据收集 的数据相匹配。

# 控制重复事件

使用事件过滤和摘要选项来控制如何将重复事件发送至 Tivoli Monitoring。

# 开始之前

有关事件过滤和摘要的更多信息,请参阅第 219 页的『第 16 章 事件过滤和摘要』。

### 关于此任务

Agent Builder 定义了属性组,这些属性组在 Tivoli Monitoring 中将事件数据表示为纯事件。这些属性组包括 日志文件、AIX 二进制日志、SNMP 事件和 JMX 通知。这些属性组可产生多个重复事件。您可以控制向 Tivoli Monitoring 发送这些重复事件的方式。在高级窗口中高级数据源属性下的事件信息选项卡中,您可以 针对日志文件、SNMP 事件和 JMX 通知属性组激活这些控件。

是否将事件视为其他事件的重复事件取决于您在属性组中定义的键特性。当事件中所有键特性的值与现有事件中相同键特性的值匹配时,发生重复事件。启用事件过滤和摘要时,会自动添加 isSummary、occurrenceCount、summaryInterval 和 eventThreshold 函数的特性。

### 过程

· 在事件过滤和摘要选项区域中,选择下列其中一个选项:

· 不进行事件过滤或摘要:发送没有任何事件过滤或摘要的所有事件。此选项为缺省选项。

**过滤并摘要事件**:为具有重复事件的每个事件以及基于键特性的每个唯一事件创建摘要记录。选择此 选项还会选择事件过滤选项。在**摘要选项**区域中,输入摘要时间间隔。您可以输入以秒为单位的值, 或插入配置属性。

事件过滤选项如下:

- 仅发送摘要事件: 仅发送指定时间间隔的摘要记录。
- 发送所有事件:发送所有事件和摘要记录。
- 发送第一个事件: 对于每个事件, 仅发送在所指定摘要时间间隔收到的第一个事件, 不发送重复事件。此选项还会发送摘要记录。
- 事件阈值:当时间间隔中收到的重复事件数可以被此阈值整除时,向 Tivoli Monitoring 发送事件。 例如,如果将事件阈值设置为5,并且在时间间隔中收到的重复事件(包括第一个事件)少于5,那 么不会向 Tivoli Monitoring 发送事件。如果收到5、6、7、8或9个重复事件,那么将发送一个事 件。如果收到10个重复事件,那么发送2个事件。在事件阈值字段中,您可以输入数字或插入配 置属性。此选项还会发送摘要记录。

# 在 Tivoli Enterprise Portal 中查看事件过滤和摘要

关于如何根据事件过滤和摘要选项处理数据的示例。

代理程序将保留最近接收到的事件的高速缓存。在缺省情况下,此高速缓存大小为 100。如果启用代理程序 事件过滤和摘要,那么高速缓存中的事件数与发送到 IBM Tivoli Monitoring 的数量可能不同。高速缓存中的 其他事件可能不会达到指定的发送阈值。或者,如果选择了**发送所有事件**选项,那么高速缓存中的事件数可 能较少。如果设置**发送所有事件**选项,那么在每次发生重复时将发送事件。但是,仅在高速缓存中保留事件 的一个副本,并且每次发生事件时出现次数都将递增。要查看发送至 IBM Tivoli Monitoring 的事件,请创建 历史视图。有关创建历史视图的信息,请参阅《Tivoli Enterprise Portal 用户指南》中的『历史报告』。您 可以将此视图与 Tivoli Enterprise Portal 中的实时高速缓存视图进行比较。还可以使用情境来进行此比较。

下列示例说明如何根据您对事件过滤和摘要进行的选择(如果有)处理相同的日志数据。创建的示例代理程序用于描述不同的行为。每个属性组都定义为监视同一个日志文件。在每个示例中都显示历史视图和实时(高速缓存)视图。Tivoli Enterprise Portal 中的节点名称反映了所选设置。在缺省情况下,历史视图将最新事件显示在最后。高速缓存的缺省实时视图将最新事件显示在最前。在这些示例中,历史视图显示过去1小时的事件。

当有新事件到达时,您会在高速缓存视图中看到这些事件。当有重复事件到达时,现有行内的数据将进行更新。经过摘要时间间隔后,现有事件将转换为摘要事件并发送。然后将为下一个摘要时间间隔添加新行。

(<u>第 221 页的图 45</u>)显示没有启用事件过滤或摘要时的历史视图和高速缓存视图。这两个视图都显示相同数据,但顺序相反。为了显示对应事件,历史视图向下滚动,而实时(高速缓存)视图向上滚动。

Jog Old Way - loc	log Old Way - localhost - SYSADMIN *ADMIN MODE*									
			A 100 A 1		~	🗛 🖬 🛱 🕅	19 <b>(1) (2)</b>			
		10 2 H   III	·							
and Navigator				± Ш E		This view has not been defined				
s 3	Vie	ew: Physical		- (	2		) % 🔄 🖨 🤇	💫 Location: 💽 http://localhost:1920///cnp/kdh.	'lib/classes/ca	
Enterprise						This viev	v has not l	been defined	<b>A</b>	
🕒 🧰 UNIX Systems	5							с. н. м. н. м. н. н. н. н. н. н. н. н.		
Vindows Sys	tems				I his is the det	lault workspace f You have this for	for this Navigator item, and no view has been never view and a table view. You can enter :			
	amnle				URL in the ad	dress text box to	o open a Web page. You can also change to			
	a Summary Only				•	a different view	v or add more vie	ews as described in these topics:		
- 💭 log	Summary And All				Hands on practi	ice and oueruieurs	View choices			
	) Old Way						ice and overviews			
- 🖳 log	3 Summary And Events 5					Tutorial: D	efining a workspace	Invoir Enterprise Console event viewer		
	Summary And First					Using worksp	paces	Table view		
Pe	rformance Object Status					Customizing	workspaces	😔 🛄 🕾 🔤 🥶 🚺 <u>Chart views</u>		
Rhysical						Done		III addies		
						Jeone				
Historical View										
Recording Time	Node	Timestamp	ID	Source		Message				
08/06/10 14:16:00	IBM-5DB67092DEE:25	08/06/10 14:16:25	INFORMATIO	N:100 Source - C	2 I	Message Text			<u> </u>	
08/06/10 14:21:00	IBM-5DB67092DEE:25	08/06/10 14:21:40	INFORMATIO	N:100 Source - 0	2	Message Text				
08/06/10 14:21:00	IBM-5DB67092DEE:25	08/06/10 14:21:40	INFORMATIO	N:100 Source - (	3 1	Message Text				
08/06/10 14:21:00	IBM-5DB67092DEE:25	08/06/10 14:21:41		N:100 Source - C	2 1	Message lext				
08/06/10 14:21:00	IBM-5DB67092DEE:25	08/06/10 14:21:43	WARNING:56	Source - E	3 1	Message Text				
08/06/10 14:21:00	IBM-5DB67092DEE:25	08/06/10 14:21:44	WARNING:56	Source - E	3 1	Message Text				
08/06/10 14:21:00	IBM-5DB67092DEE:25	08/06/10 14:21:45	WARNING:56	Source - E	3 1	Message Text				
08/06/10 14:21:00	IBM-5DB67092DEE:25	08/06/10 14:21:45	WARNING:56	Source - E	3 1	Message Text				
08/06/10 14:21:00	IBM-5DB67092DEE:25	08/06/10 14:21:46	WARNING:56	Source - E	3 1	Message Text				
08/06/10 14:21:00	IBM-5DB67092DEE:25	08/06/10 14:21:46	WARNING:56	Source - E	3	Message Text				
08/06/10 14:21:00	IBM-5DB67092DEE:25	08/06/10 14:21:47	WARNING:56	Source - E	3 1	Message Text				
08/06/10 14:21:00	IBM-5DB67092DEE:25	08/06/10 14:21:47	WARNING:56	Source - E	5 1	Message Text				
08/06/10 14:21:00	IBM-5DB67092DEE:25	08/06/10 14:21:48	WARNING:56	Source - E	3 1	Message Text				
© Last 1 Hours.				1		,				
Cache View					-			/ \$ []	8 0 ×	
Node	Timestamn	ID	Source	Message						
IBM-5DB67092DEE:	25 08/06/10 14:21:48	WARNING:56	Source - B	Message Text					-	
IBM-5DB67092DEE:	25 08/06/10 14:21:48	WARNING:56	Source - B	Message Text						
IBM-5DB67092DEE:	25 08/06/10 14:21:47	WARNING:56	Source - B	Message Text						
IBM-5DB67092DEE:	25 08/06/10 14:21:47	WARNING:56	Source - B	Message Text						
IBM-5DB67092DEE:	25 08/06/10 14:21:46	WARNING:56	Source - B	Message Text						
IBM-5DB67092DEE:	25 08/06/10 14:21:46	WARNING:56	Source - B	Message Text						
IBM-5DB67092DEE:	25 08/06/10 14:21:45	WARNING:56	Source - B	Message Text						
IBM-5DB67092DEE.	25 08/06/10 14:21:45	WARNING:56	Source - B	Message Text						
IBM-5DB67092DEE:	25 08/06/10 14:21:44	WARNING:56	Source - B	Message Text						
IBM-5DB67092DEE:	DEE:25 08/06/10 14:21:43 WARNING:56 Source - B Message Text		Message Text							
IBM-5DB67092DEE:	2DEE:25 08/06/10 14:21:41 INFORMATION:100 Source - Q Message Text		Message Text							
IBM-5DB67092DEE:	25 08/06/10 14:21:40	INFORMATION:100	Source - Q	Message Text						
IBM-5DB67092DEE:	25 08/06/10 14:21:40	INFORMATION:100	Source - Q	Message Text						
IBM-5DB67092DEE:	25 08/06/10 14:16:25	INFORMATION:100	Source - Q	Message Text						
IBM-5DB67092DEE:	25 08/06/10 14:16:25	INFORMATION:100	Source - Q	wessage rext					<u></u>	
🕒 Hub Time: Fri, 08/06/2010 02:22 PM 🔇 Server Available							log Old Way - I	localhost - SYSADMIN *ADMIN MODE*		

图 45. 未启用事件过滤或摘要时的历史视图和高速缓存视图

<u>第 222 页的图 46</u>显示在"事件信息"选项卡中选择了仅发送摘要事件选项时的历史视图和高速缓存视图。 摘要事件将同时显示在这两个视图中,但新事件只显示在实时(高速缓存)视图中。

log Summary Only - localhost - SYSADMIN *ADMIN MODE*													
<u>File Edit View H</u> elp	p												
♠ ♀ • • • 1	I 🖬 🔛 🖉 🥸 🛽	80 🖷 💷	ې 📰 🗞 🥘	\$ 🛛 🕙 🛄	lo 🕾 🛙	o 🗗 🖬 🖬 🖉 🖳 🔗 🖳 🛍 🛄 👘 💿							
😪 Navigator				\$		🔀 This vie	🕼 This view has not been defined 🛛 🖈 💷 🖯 🗙						
* 🧭	Vie	ew: Physical			- 0	🔒 🏠 🍁 📫	· 🔵 🖑 🥥 🖨 🤇	🖕 Location: 💽 htt	p://localhost:1920///cn	p/kdh/lib/classes.	sica		
Enterprise						Thie v	iew hae not l	aan dafina	d		*		
🕒 🛅 UNIX Systems						11115 V	iew nas not i	Jeen denne	u				
🕒 🚞 Windows System	ms					This is the	default workspace f	or this Navigator i	tem, and no view has	s been			
🖹 🗐 🏥 IBM-5DB670	92DEE					defined he	re. You have this bro	wserview and a t	able view. You can e	entera			
🖃 🙋 LogExan	nple					I a different	view or add more vie	ws as described i	n these topics:	nge to			
	Summary Only												
	ummary And All					Hands-on p	ractice and overviews	View choices					
	Summary And Events 5					Tutor	al: Defining a workspace	💕 <u>Tivoli Enterprise</u>	Console event viewer				
- 📮 log 8	ummary And First					🖽 Using u	orkspaces	Table view					
📃 💭 Perfo	ormance Object Status					Custor	izing workspaces	3 11 🕾 🚔 🖲	Chart views				
								<u>شا</u> ر			-		
Physical	Done												
Historical View	🔲 Historical View / 🏦 💷 🖯 🗙												
D Q													
Recording Time	Node	Timestamp	ID	Source		Message	Occurrence Cour	t Event Type	Summary Interval	Event Threshold	d		
08/06/1014:03:00 IE	3M-5DB67092DEE:25	08/06/10 14:03:36	INFORMATION	N:100 So	urce - Q	Message Tex	t 3	Summary Event	120	SEND NONE			
08/06/10 14:03:00 IE	3M-5DB67092DEE:25	08/06/10 14:03:36	WARNING:56	Source -		Source - B		Message Te	t 2	Summary Event	: 120	SEND NONE	
08/06/1014:17:00 IE	3M-5DB67092DEE:25	08/06/10 14:17:36	INFORMATION	V:100 Source - Q		Message Tex	t 3	Summary Event	: 120	SEND NONE	_		
08/06/1014:17:00 IE	3M-5DB67092DEE:25	08/06/10 14:17:36	WARNING:56	3 Source - B		Message Tex	t   5	Summary Event	:  120	SEND NONE			
🕒 Last 1 Hours.													
Cache View									1 :		×		
Node	Timestamp	ID	Source	Messag	ge C	)ccurrence Cour	it Event Type	Summary Interval	Event Threshold				
IBM-5DB67092DEE:25	08/06/10 14:21:43	WARNING:56	Source - B	Message "	Text 1	1	Event	120	SEND NONE				
IBM-5DB67092DEE:25	08/06/10 14:21:40	INFORMATION:100	Source - Q	Message <sup>-</sup>	Text 3	1	Event	120	SEND NONE				
IBM-5DB67092DEE:25	08/06/10 14:17:36	INFORMATION:100	Source - Q	Message Text			Summary Event	120	SEND NONE				
IBM-5DB67092DEE:25	08/06/10 14:17:36	WARNING:56	Source - B	Message Text		i i i i i i i i i i i i i i i i i i i	Summary Event	120	SEND NONE				
IBM-5DB67092DEE:25	08/06/10 14:03:36	INFORMATION:100	Source - Q	Message Text 3			Summary Event	120	SEND NONE				
IBM-5DB67092DEE:25	08/06/10 14:03:36	WARNING:56	Source - B	message ]	rext 2		Summary Event	120	SEND NONE				
	Hub Time: Fri, 08/06/2010 02:21 PM Server Available Iog Summary Only - localhost - SYSADMIN *ADMIN MODE*												

图 46. 选择了**仅发送摘要事件**时的历史视图和高速缓存视图

<u>第223</u>页的图 47 显示在"**事件信息**"选项卡中选择了**发送所有事件**选项时的历史视图和高速缓存视图。两 个视图都显示所有事件,但您还可以看到每个时间间隔结束时创建的摘要事件。实时视图随时间间隔流逝而 变化。现有事件将转换为摘要记录,然后添加新事件。还添加了另外两个可用的事件属性,这两个属性用于 显示摘要时间间隔(在本例中为 120 秒)和 SEND ALL 阈值。

Iog Summary And File Edit View He	l All - localhost - SYSADM alp	IIN *ADMIN MODE*							<u>_</u> _×
<b>☆・</b> ♦ • [	] 🖬 🛛 🕾 🕹 🖸	80 @	۵ 😵 🖷 🗞	🛛 🕙 🛄 😤	🚔 😬 🔟 🗒 🕻	1 🛛 🗟 🖻 🤗	📮 🛃 🚠 🚺	<b>1</b>	5
🗠 Navigator				â 🔟 E	🗧 🔣 This view	has not been define	đ	1 :	
* 1	Vie	ew: Physical			💿 🖑 🗳 🖨 🔍	Location: 💽 http	://localhost:1920///cn	p/kdh/lib/classes/ca	
Enterprise	ems			This vie	ew has not b	een define	d em, and no view has	been	
E S LogExa	ample Summary Only Summary And All Old Way			URL in the a a different vi	URL in the address text box to open a Web page. You can also change to a different view or add more views as described in these topics: Hands-on practice and overviews View choices				
Peri	Summary And Events 5 Summary And First formance Object Status			Using wor	Tutorial: Defining a workspace     Using workspaces     Using workspaces     Customizing workspaces     Customizing workspaces     Customizing workspaces     Customizing workspaces     Customizing workspaces				
Physical					Done				
Historical View								1 :	
Recording Time	Node	Timestamn	ID	Source	Message	Occurrence Count	Event Type	Summary Interval	Event Threshold
08/06/10 14:16:00	IBM-5DB67092DEE:25	08/06/10 14:16:19	WARNING:56	Source -	B Message lext	1	Event	120	SEND ALL
08/06/10 14:16:00	IBM-5DB67092DEE:25	08/06/10 14:16:19	WARNING:56	Source -	B Message Text	1	Event	120	SEND ALL
08/06/10 14:16:00	IBM-5DB67092DEE:25	08/06/10 14:16:20	WARNING:56	Source -	B Message Text	1	Event	120	SEND ALL
08/06/10 14:16:00	IBM-5DB67092DEE:25	08/06/10 14:16:21	WARNING:56	Source -	B Message Text	1	Event	120	SEND ALL
08/06/10 14:16:00	IBM-5DB67092DEE:25	08/06/10 14:16:24	INFORMATION:1	00 Source -	Q Message Text	1	Event	120	SEND ALL
08/06/10 14:16:00	IBM-5DB67092DEE:25	08/06/10 14:16:25	INFORMATION:1	00 Source -	Q Message Text	1	Event	120	SEND ALL
08/06/10 14:16:00	IBM-5DB67092DEE:25	08/06/10 14:16:25	INFORMATION:1	00 Source -	Q Message Text	1	Event	120	SEND ALL
08/06/10 14:17:00	IBM-5DB67092DEE:25	08/06/10 14:17:36	INFORMATION:1	00 Source -	Q Message Text	3	Summary Event	120	SEND ALL
08/06/10 14:17:00	IBM-5DB67092DEE:25	08/06/10 14:17:36	WARNING:56	Source -	B Message Text	5	Summary Event	120	SEND ALL
08/06/10 14:21:00	IBM-5DB67092DEE:25	08/06/10 14:21:40	INFORMATION:1	00 Source -	Q Message Text	1	Event	120	SEND ALL
08/06/10 14:21:00	IBM-5DB67092DEE:25	08/06/10 14:21:40	INFORMATION:1	00 Source -	Q Message Text	1	Event	120	SEND ALL
08/06/10 14:21:00	IBM-5DB67092DEE:25	08/06/10 14:21:41	INFORMATION:1	00 Source -	Q Message Text	1	Event	120	SEND ALL
08/06/10 14:21:00	IBM-5DB67092DEE:25	08/06/10 14:21:43	WARNING:56	Source -	B Message Text	1	Event	120	SEND ALL
08/06/10 14:21:00	IBM-5DB67092DEE:25	08/06/10 14:21:44	WARNING:56	Source -	B Message Text	1	Event	120	SEND ALL
08/06/10 14:21:00	IBM-5DB67092DEE:25	08/06/10 14:21:44	WARNING:56	Source -	B Message Text	1	Event	120	SEND ALL
08/06/10 14:21:00	IBM-5DB67092DEE:25	08/06/10 14:21:45	WARNING:56	Source -	B Message Text	1	Event	120	SEND ALL
08/06/10 14:21:00	IBM-5DB67092DEE:25	08/06/10 14:21:45	WARNING:56	Source -	B Message Text	1	Event	120	SEND ALL
08/06/10 14:21:00	IBM-5DB67092DEE:25	08/06/10 14:21:46	WARNING:56	Source -	B Message Text	1	Event	120	SEND ALL
08/06/10 14:21:00	IBM-5DB67092DEE:25	08/06/10 14:21:46	WARNING:56	Source -	B Message Text	1	Event	120	SEND ALL
08/06/10 14:21:00	IBM-5DB67092DEE:25	08/06/10 14:21:47	WARNING:56	Source -	B Message Text	1	Event	120	SEND ALL
08/06/10 14:21:00	IBM-5DB67092DEE:25	08/06/10 14:21:47	WARNING:56	Source -	B Message Text	1	Event	120	SEND ALL
08/06/10 14:21:00	IBM-5DB67092DEE:25	08/06/10 14:21:48	WARNING:56	Source -	B Message Text	1	Event	120	SEND ALL
08/06/10 14:21:00	IBM-5DB67092DEE:25	08/06/10 14:21:48	WARNING:56	Source -	B Message Text	1	Event	120	SEND ALL
									<u></u>
G Last 1 Hours.									
Cache View								/ :	
Node	Timestamp	ID	Source	Messade	Occurrence Count	Event Type 8	Summary Interval	Event Threshold	
IBM-5DB67092DEE:2	5 08/06/10 14:21:43	WARNING:56	Source - B M	essage Text	11	Event 1	20	SEND ALL	
IBM-5DB67092DEE:2	5 08/06/10 14:21:40	INFORMATION:100	Source - Q M	essage Text	3	Event 1	20	SEND ALL	
IBM-5DB67092DEE:2	5 08/06/10 14:17:36	INFORMATION:100	Source - Q M	essage Text	3	Summary Event 1	20	SEND ALL	
IBM-5DB67092DEE:2	5 08/06/10 14:17:36	WARNING:56	Source - B M	essage Text	5	Summary Event 1	20	SEND ALL	
IBM-5DB67092DEE:2	5 08/06/10 14:03:36	INFORMATION:100	Source - Q M	essage Text	3	Summary Event 1	20	SEND ALL	
IBM-5DB67092DEE:2	5 08/06/10 14:03:36	WARNING:56	Source - B M	essage Text	2	Summary Event 1	20	SEND ALL	
	Hub Time: Fri, 08/06/2010 02:22 PM     Server Available     log Summary And All - localhost - SYSADMIN *ADMIN MODE*								

图 47. 选择了**发送所有事件**时的历史视图和高速缓存视图

<u>第 224 页的图 48</u>显示在"**事件信息**"选项卡中选择了**发送第一个事件**选项时的历史视图和高速缓存视图。 摘要事件将同时显示在这两个视图中,但所有新事件只显示在实时(高速缓存)视图中。对于每个事件,历 史视图仅显示在时间间隔内收到的第一个事件,不显示重复事件。

📕 log Summary And First - localhost - SYSAI	OMIN *ADMIN MODE	•							
<u>File Edit View H</u> elp									
♠ 🗇 • ♠ •   🐌 🖬   🔢 🖉 😵 🛙		🥥 🛷 📰 🖏	🛛 🕙 🛄 😤	🚔 😬 🔟 🗒 🕻	1 🛛 🖓 🖻 🔗	📮 🗖 🚓 🚺 🖡		5	
😪 Navigator			\$ 00 E	🗌 🔣 This view	has not been define	1	1 :		
🔹 📝 🗸 👔	ew: Physical		-	۵ 💠 🔿	o 🚸 🕒 🖨 🔍	Location: 💽 http	://localhost:1920///cn	p/kdh/lib/classes/ca	
Enterprise					w bac not b	oon dofino	ч		
ONIX Systems     O			This is the c defined here URL in the a a different vi	This is the default workspace for this Navigator item, and no view has been defined here. You have this <i>browser view</i> and a <i>table view</i> . You can enter a URL in the address text box to open a Web page. You can also change to a different view or add more views as described in these topics:					
- 🖳 log Summary And Events 5				Tutonal	: Defining a workspace		console event viewer		
log Summary And First				Using wor	kspaces	Table view			
– 🖳 Performance Object Status				Customizi	ng workspaces		Chart views		
						Adding a notepad	view	-	
et Physical						<u> </u>	1 - 1		
				IDoue					
🔲 Historical View 🥒 🖈 💷 🖯 📼 🖈									
Recording Time Node	Timestamp	ID	Source	Message	Occurrence Count	Event Type	Summary Interval	Event Threshold	
08/06/10 14:02:00 IBM-5DB67092DEE:25	08/06/10 14:02:45	WARNING:56	Source - E	3 Message Text	1	Event	120	SEND FIRST	
08/06/10 14:02:00 IBM-5DB67092DEE:25	08/06/10 14:02:54	INFORMATION	:100 Source - (	Message Text	1	Event	120	SEND FIRST	
08/06/10 14:03:00 IBM-5DB67092DEE:25	08/06/10 14:03:36	INFORMATION	100 Source - C	Message Text	3	Summary Event	120	SEND FIRST	
08/06/1014:03:00 IBM-5DB67092DEE:25	08/06/10 14:03:36	WARNING:56	Source - E	3 Message Text	2	Summary Event	120	SEND FIRST	
08/06/10 14:16:00 IBM-5DB67092DEE:25	08/06/10 14:16:18	WARNING:56	Source - E	3 Message Text	1	Event	120	SEND FIRST	
08/06/10 14:16:00 IBM-5DB67092DEE:25	08/06/10 14:16:24	INFORMATION	100 Source - C	Message Text	1	Event	120	SEND FIRST	
08/06/10 14:17:00 IBM-5DB67092DEE:25	08/06/10 14:17:36	INFORMATION	100 Source - C	Message Text	3	Summary Event	120	SEND FIRST	
08/06/10 14:17:00 IBM-5DB67092DEE:25	08/06/10 14:17:36	WARNING:56	Source - E	3 Message Text	5	Summary Event	120	SEND FIRST	
08/06/10 14:21:00 IBM-5DB67092DEE:25	08/06/10 14:21:40	INFORMATION	100 Source - C	Message Text	1	Event	120	SEND FIRST	
08/06/10 14:21:00 IBM-5DB67092DEE:25	08/06/10 14:21:43	WARNING:56	Source - E	3 Message Text	1	Event	120	SEND FIRST	
08/06/1014:23:00 IBM-5DB67092DEE:25	08/06/10 14:23:36	WARNING:56	Source - E	3 Message Text	11	Summary Event	120	SEND FIRST	
08/06/1014:23:00 IBM-5DB67092DEE:25	08/06/10 14:23:36	INFORMATION	100 Source - C	Message Text	3	Summary Event	120	SEND FIRST	
08/06/10 14:24:00 IBM-5DB67092DEE:25	08/06/10 14:24:06	WARNING:56	Source - E	3 Message Text	1	Event	120	SEND FIRST	
08/06/10 14:24:00   IBM-5DB67092DEE:25	08/06/10 14:24:10	INFORMATION	:100   Source - (	Message Text	1	Event	120	SEND FIRST	
© Last 1 Hours.									
		1 - 1							
Node Timestamp	ID	Source	Message	Occurrence Count	Event Type 8	Summary Interval	Event Threshold		
IBM-5DB67092DEE:25 08/06/10 14:24:10	INFORMATION:100	Source - Q	Message lext	3	Event 1	20	SENDFIRST		
IBM-5DB67092DEE:25 08/06/10 14:24:06	WARNING:56	Source - B	Message lext	b 4.4	Event 1	20	SEND FIRST		
IBM-5DB67092DEE:25 08/06/10 14:23:36	WARNING:56	Source - B	Message lext	11	Summary Event 1	20	SEND FIRST		
IBM-5DB67092DEE:25 U8/06/10 14:23:36	INFORMATION:100	Source - Q	wessage lext	J 0	Summary Event 1	20	SEND FIRST		
IBM-5DB67092DEE:25 08/06/10 14:17:36	JDB67092DEE:25 08/06/10 14:17:36 INFORMATION:100 Source - Q Message Text 3		3	Summary Event	20	SEND FIRST			
IBM-5DB67092DEE:25 08/06/10 14:17:36	WARINING:56	Source - B	wessage rext	5 2	Summary Event 1	20	SENU FIRST		
IBM-5DB67092DEE:25 08/06/10 14:03:36	INFORMATION:TUU	Source - Q	Message rext	3	Summary Event 1	20	SENU FIRST		
IBW-508670920EE.25 08/06/1014:03:36	WARNING:56	aource - B	wessage rext	2	ournmary Event 1	20	SEND FIRST		
🕒 Hub Time: Fri, 08/06/	Hub Time: Fri, 08/06/2010 02:24 PM     Server Available     log Summary And First - localhost - SYSADMIN *ADMIN MODE*								

图 48. 选择了发送第一个事件时的历史视图和高速缓存视图

<u>第225页的图49</u>显示选择了**事件阈值**选项并输入值5时的历史视图和高速缓存视图。摘要事件将同时显示 在这两个视图中,但所有新事件只显示在实时(高速缓存)视图中。本示例中指定的阈值为5。仅当在时间 间隔内接收到一个事件的五个重复事件(包括第一个事件)时,历史视图才会显示事件。如果接收到的重复 事件不足5个,那么不显示任何事件。如果在此时间间隔内收到6、7、8或9个重复事件,那么只显示一 个事件。如果收到10个重复事件,那么显示2个事件。

log Summary And Events 5 - localhost - SYSADMIN *ADMIN MODE*											
<u>File Edit View H</u> e	lp										
<b>☆・</b> • • [	] 🖬 🔛 🛃 🥸 🛽		🥥 🖑 🖽 🍕	k   🎯 🌆 😤		•	1 1 1 2 9	📕 🗖 🎄 🚺			
🗠 Navigator				<b>\$</b> []]	8	🛃 This view	has not been define	ed	1		□ ×
* 🧭	Vie	ew: Physical		•	Q	☆ 🗭 🔿	🔵 🖑 🥥 😓 🔾	🕻 Location: 💽 http	o://localhost:1920///cn	p/kdh/lib/cla	asses/ca
Enterprise  UNIX Systems  Windows Syste  UNIX Systems  U	-	This view has not been defined         This is the default workspace for this Navigator item, and no view has been defined here. You have this <i>browser view</i> and a <i>table view</i> . You can enter a URL in the address text box to open a Web page. You can also change to a different view or add more views as described in these topics:         Hands-on practice and overview       Vew obcices         Using workspaces       Involt Enterprise Console event viewer         Using workspaces       Involt Enterprise Console event viewer         Outprinting workspaces       Involt Enterprise Console event viewer					×				
A Physical	C Physical Done										
🔲 Historical View	🛄 Historical View / 🏦 💷 🗄 🗆 🗴										
D 🔍											
Recording Time	Node	Timestamp	ID	Sourc	e	Message	Occurrence Cour	t Event Type	Summary Interval	Event Thre	eshold
08/06/10 14:03:00	IBM-5DB67092DEE:25	08/06/10 14:03:36	INFORMATION	I:100 Source	- Q	Message Text	3	Summary Event	120	5	
08/06/10 14:03:00	IBM-5DB67092DEE:25	08/06/10 14:03:36	WARNING:56	Source	B	Message Text	2	Summary Event	120	5	
08/06/10 14:16:00	IBM-5DB67092DEE:25	08/06/10 14:16:21	WARNING:56	Source	·В	Message Text	1	Event	120	5	
08/06/10 14:17:00	IBM-5DB67092DEE:25	08/06/10 14:17:36	INFORMATION	I:100 Source	Q	Message Text	3	Summary Event	120	5	
08/06/10 14:17:00	IBM-5DB67092DEE:25	08/06/10 14:17:36	WARNING:56	Source -	- B	Message Text	5	Summary Event	120	5	
08/06/10 14:21:00	IBM-5DB67092DEE:25	08/06/10 14:21:45	WARNING:56	Source	- B	Message Text	1	Event	120	5	
08/06/10 14:21:00	IBM-5DB67092DEE:25	08/06/10 14:21:48	WARNING:56	Source -	·В	Message Text	1	Event	120	5	
© Last 1 Hours. □ Cache View / ★ □ 日 □ ×											
Node	Timestamp	ID	Source	Message	Occ	urrence Count	Event Type	Summary Interval	Event Threshold		
IBM-5DB67092DEE:2	5 08/06/10 14:21:43	WARNING:56	Source - B	Message Text	11		Event	120	5		
IBM-5DB67092DEE:2	5 08/06/10 14:21:40	INFORMATION:100	Source - Q	Message Text	3		Event	120	5		
IBM-5DB67092DEE:2	5 08/06/10 14:17:36	INFORMATION:100	Source - Q	Message Text	3		Summary Event	120	5		
IBM-5DB67092DEE:2	5 08/06/10 14:17:36	WARNING:56	Source - B	Message Text	5		Summary Event	120	5		
IBM-5DB67092DEE:2	5 08/06/10 14:03:36	INFORMATION:100	Source - Q	Message Text	3		Summary Event	120	5		
IBM-5DB67092DEE:2	5 08/06/10 14:03:36	WARNING:56	Source - B	Message Text	2		Summary Event	120	5		
0	C Hub Time: Fri, 08/06/2010 02:23 PM Server Available log Summary And Events 5 - localhost - SYSADMIN *ADMIN MODE*										

图 49. 选择了事件阈值时的历史视图和高速缓存视图

# 相关概念

第219页的『事件过滤和摘要』

# 第17章 故障诊断和支持

请查看故障诊断信息,以了解您在安装、配置或使用 IBM Agent Builder 时可能遇到的问题。

有关在 IBM Cloud Application Performance Management 环境中开发、安装或使用定制代理程序时的问题 故障诊断帮助,请参阅 developerWorks<sup>®</sup>上的 <u>Cloud Application Performance Management 论坛</u>。您可以 搜索 "agent\_builder"标记,回复提出相关问题的条目,或使用您的问题创建新条目。

有关日志记录和消息参考信息以及与 IBM Tivoli Monitoring 环境有关的问题的故障诊断帮助,请参阅 IBM Agent Builder 故障诊断参考。

# 附录 A 共享项目文件

与他人共享 IBM Tivoli Monitoring 代理程序项目。

# 过程

- 获取文件。您需要与工作空间目录中的项目同名目录的全部内容。 例如,如果工作空间目录为c:\Documents and Settings\User1\workspace,并且您想要共享名 为TestProject的项目。必须使目录c:\Documents and Settings\User1\workspace \TestProject及其所有内容可供系统访问。
- 2. 选择文件 > 导入。
- 3. 打开 IBM Tivoli Monitoring。
- 4. 选择 IBM Tivoli Monitoring Agent 并单击下一步。
- 5. 输入代理程序 XML 文件的完整路径,或者单击 浏览以浏览到该文件。
- 6. 单击**完成**。

# 结果

在向导完成时,您将会在工作空间中看到新的 IBM Tivoli Monitoring 代理程序项目。

# 共享解决方案安装程序项目

与他人共享解决方案安装程序项目

# 过程

- 获取文件。您必须具有与工作空间目录中解决方案安装程序项目同名的目录的全部内容。 例如,如果工作空间目录为c:\Documents and Settings\User1\workspace,并且您想要共享名 为TestProject Installer的解决方案安装程序项目。必须使目录c:\Documents and Settings\User1\workspace\TestProject Installer 及其所有内容可供系统访问。
- 2. 单击**文件 > 导入**。
- 3. 打开**常规**。
- 4. 选择现有项目到工作空间,并单击下一步。
- 5. 输入解决方案安装程序项目根目录的完整路径,或者单击**浏览**以浏览到解决方案安装程序项目的根目 录。(在此示例中为 TestProject Installer 目录。)该目录中的项目显示在 Projects 列表中并且 缺省情况下处于选中状态。
- 6. 可选: 单击将项目复制到工作区中。
- 7. 单击**完成**。

# 附录 B 命令行选项

Agent Builder 命令行界面 (CLI) 中可用的命令。

Tivoli Monitoring Agent Builder 包含命令行界面 (CLI),您可以使用该界面来生成 Tivoli Monitoring Agent,而无需启动 Eclipse 图形用户界面 (GUI)。可以作为构建的一部分而生成代理程序,例如:

在 Windows 系统上,可以使用下列目录中的批处理文件来访问 CLI:

install\_location\agenttoolkit.bat

在 UNIX 及 Linux 系统上,可以使用下列目录中的脚本来访问 CLI:

install\_location/agenttoolkit.sh

本文档中描述的命令针对 Windows 系统编排格式,在这些系统上,目录路径使用反斜杠 (\)。

对于 UNIX<sup>®</sup> 或 Linux<sup>®</sup> 系统,请使用与 Windows 系统相同的命令,但是进行下列更改:

·在目录路径中,请使用正斜杠 (/),而不要使用反斜杠 (\)。

·请使用 agenttoolkit.sh 脚本,而不要使用 agenttoolkit.bat 脚本。

### 命令

第231页的表44列出文本命令的每个命令选项的名称及用途说明:

表 44. 命令快速参照表							
命令	用途						
generatelocal	装入并验证 itm_toolkit_agent.xml 文件,并生成用于运行 Tivoli Monitoring Agent 的文件。将会安装至本地 Tivoli Monitoring 环境。						
generatemappingfile	创建映射文件,用于将定制 IBM Tivoli Monitoring V5.x 资源模型移植至 IBM Tivoli Monitoring V6 代理程序。						
generatezip	生成名为 productcode.zip 或 productcode.tgz 的压缩文件。						

从表中引用的命令描述说明了如何运行命令,其中包含下列信息:

### 用途

列示命令的用途。

# 格式

指定在命令行输入的语法。语法包含命令名称和命令的参数列表。在命令名称之后,跟着每个参数的定义。

# 示例

命令的示例包含示例的简短描述和语法示例。

# 用法

提供命令的说明及其用法。

# 注释

提供能够传达更多信息的命令或文本。

# 命令 - generatelocal

使用此命令可以装入并验证 XML,并生成用于运行 Tivoli Monitoring Agent 的文件。

### 用途

装入并验证 itm\_toolkit\_agent.xml 文件,并生成用于运行 Tivoli Monitoring Agent 的文件。将会安装 至本地 Tivoli Monitoring 环境。

### 格式

对于 Windows 系统:

install\_location\agenttoolkit.bat project\_dir -generatelocal itm\_install\_dir

其中:

*install\_location* Agent Builder 的安装目录

#### project\_dir

itm\_toolkit\_agent.xml 文件所在目录的名称

#### itm\_install\_dir

Tivoli Monitoring 的安装位置(例如 c:\IBM\ITM)

### 示例

在下列 Windows 示例中,将验证 C:\ABCAgent 中的代理程序定义,且会在 C:\IBM\ITM 中生成运行 ABCAgent 所需的文件:

install\_location\agenttoolkit.bat C:\ABCAgent -generatelocal C:\IBM\ITM

# 命令 - generatemappingfile

使用此命令可以将定制 IBM Tivoli Monitoring V5.x 资源模型迁移至 IBM Tivoli Monitoring V6 代理程序。

# 用途

此命令会创建映射文件,用于将定制 IBM Tivoli Monitoring V5.x 资源模型迁移至 IBM Tivoli Monitoring V6 代理程序。

### 格式

对于 Windows 系统:

```
install_location\agenttoolkit.bat project_dir -generatemappingfile output_dir
    itm5_interp_list
```

其中:

# install\_location Agent Builder 的安装目录 project dir

itm\_toolkit\_agent.xml 所在目录的名称

output\_dir

映射文件所写入的目录的名称

### itm5\_interp\_list

运行定制资源模型的 ITM 5x 操作系统的逗号分隔列表。允许的值如下所示:

- ∙aix4-r1
- •hpux10
- ·linux-ix86
- ・linux-ppc
- ·linux-s390
- ∙os2-ix86
- ∙os400
- •solaris2
- •solaris2-ix86
- ·w32-ix86

# 示例

对于 Windows 系统

```
install_location\agenttoolkit.bat c:\ABCAgent -generatemappingfile c:\output
linux-ix86,linux-ppc,linux-s390
```

# 命令 - generatezip

使用此命令可以装入并验证 XML 以及生成可用于在其他系统上安装代理程序的压缩文件。

# 用途

装入并验证 itm\_toolkit\_agent.xml 文件以及生成名为 productcode.zip 或 productcode.tgz 的 压缩文件。所生成的压缩文件可用于在其他系统上安装代理程序。根据您的环境,可能会同时生成这两种类 型的文件。

# 格式

对于 Windows 系统:

```
install\_location \verb+ agenttoolkit.bat project\_dir - generatezip output\_dir
```

其中:

# project\_dir

itm\_toolkit\_agent.xml 文件所在目录的名称

### output\_dir

压缩文件所写入的目录的名称

# 示例

在下列 Windows 示例中,将验证 C:\ABCAgent 中的代理程序定义,且会在 C:\Output 中生成压缩文件,其中包含运行 ABCAgent 所需的文件:

install\_location\agenttoolkit.bat\ C:\ABCAgent -generatezip C:\Output

# 附录 C 属性参考

提供 Agent Builder 中每一个生成的属性组的属性描述。

# 可用性节点

"可用性"属性组包含应用程序的可用性数据。

此表提供了用于表示应用程序可用性的通用格式,包括应用程序以下三个方面的相关信息:服务(仅限 Windows)、进程和命令返回码。

以下列表包含关于"可用性"属性组中每个属性的信息:

"节点"属性-此属性是键属性

# 描述

代理程序的受管系统名称

### 类型

字符串

# 名称

属性名称 Node 列名 ORIGINNODE

"时间戳记"属性

描述

收集数据时代理程序的本地时间

类型

# 时间

名称

属性名称

Timestamp

列名 TIMESTAMP

"应用程序组件"属性 - 此属性是键属性

# 描述

应用程序某个组成部分的描述性名称

# 类型

字符串

# 名称

属性名称 Application\_Component 列名 COMPONENT "名称"属性

### 描述

进程、服务或功能测试的名称。此名称与进程可执行文件名、服务短名称或用于测试应用程序的进程的名称相匹配。

类型

字符串

名称

属性名称

Name

列名 NAME

"状态"属性

#### 描述

应用程序组件的状态。

·对于进程,值为UP、DOWN、WARNING或PROCESS\_DATA\_NOT\_AVAILABLE。对于一个进程,如 果匹配的进程正在运行,但无法收集该进程的资源使用信息,那么将对该进程显示 PROCESS\_DATA\_NOT\_AVAILABLE。

·对于服务,值为 UP、DOWN 或 UNKNOWN。如果未安装该服务,那么将显示 UNKNOWN。

·对于命令返回码,值为 PASSED 或 FAILED。

#### 类型

字符串

### 名称

属性名称

Status

# 列名

STATUS

# "全名"属性

#### 描述

进程的全名,其中包含依赖于进程的信息。此名称可能包含完整路径(如果进程是以这种方式启动)。此名称也可以包含不完整的路径,甚至可以包含进程更改过的路径。

### 类型

字符串

### 名称

属性名称

Full\_Name

列名

# FULLNAME

"类型"属性

#### 描述

指示应用程序组件的类型。组件是进程、服务或命令返回码。

#### 类型

整数(标尺)

### 名称

属性名称 Type
列名

TYPE

"虚拟大小"属性

#### 描述

进程的虚拟大小 (MB)

#### 类型

整数(标尺)

### 名称

属性名称 Virtual\_Size

列名

VIRTSIZE

"每秒的缺页故障数"属性

### 描述

进程的缺页故障率,以每秒故障数测量。此值仅包含对进程有效的数据。 **类型** 整数(标尺)

### 名称

属性名称 Page\_Faults\_Per\_Sec

### 列名

PAGEFAULTS

### "工作集大小"属性

### 描述

进程的工作集大小 (MB)。此值仅包含对进程有效的数据。

### 类型

整数(标尺)

### 名称

属性名称

Working\_Set\_Size

### 列名

WORKSET

"线程计数"属性

### 描述

此进程当前分配的线程数。此值仅包含对进程有效的数据。

类型

整数 (标尺)

### 名称

属性名称 Thread\_Count 列名 THREADS

### PID 属性

描述

与进程相关联的进程标识。此值仅包含对进程有效的数据。

类型

整数 (标尺)

名称

属性名称

PID 列名

PID

"特权时间百分比"属性

### 描述

此进程用于进行特权操作的可用处理器时间百分比

类型

整数(标尺)

### 名称

属性名称 Percent\_Privileged\_Time 列名

PERCPRIV

"用户方式时间百分比"属性

### 描述

此进程用于进行用户方式操作的可用处理器时间百分比

#### 类型

整数 (标尺)

### 名称

属性名称 Percent\_User\_Mode\_Time 列名 PERCUSER

"处理器时间百分比"属性

### 描述

此进程使用处理器来运行指令所耗用的时间百分比 **类型** 

ェ 整数(标尺)

### 名称

属性名称 Percent\_Processor\_Time

列名

PERCPROC

### "命令行"属性

### 描述

启动进程时在命令行中指定的程序名和任何自变量。在运行服务或功能测试时,此属性的值为 N/A。 **类型** 

字符串

名称

属性名称

Command\_Line

列名

"功能测试状态"属性

描述

功能测试的返回码。受监视的应用程序正常运行时,将返回 SUCCESS。当该应用程序未正常运行时,将返回 NOT\_RUNNING。当该行并非表示功能测试时,将返回 N/A。

类型

具有枚举值的整数。在 Tivoli Enterprise Portal 中将显示字符串,仓库和查询将返回数字。已定义的 值为: N/A(1)、SUCCESS(0)、GENERAL\_ERROR(2)、WARNING(3)、NOT\_RUNNING (4)、DEPENDENT\_NOT\_RUNNING(5)、ALREADY\_RUNNING(6)、PREREQ\_NOT\_RUNNING (7)、TIMED\_OUT(8)、DOESNT\_EXIST(9)、UNKNOWN(10)、 DEPENDENT\_STILL\_RUNNING(11)或INSUFFICIENT\_USER\_AUTHORITY(12)。任何其他值 在 Tivoli Enterprise Portal 中都显示数字值。

### 名称

### 属性名称

Functionality\_Test\_Status

列名

FUNCSTATUS

"功能测试消息"属性

```
描述
```

与"功能测试状态"对应的文本消息。此属性只对命令返回码有效。

类型

字符串

### 名称

属性名称

Functionality\_Test\_Message

列名

FUNCMSG

# Performance Object Status 节点

使用 Performance Object Status 属性组可以查看组成此代理程序的所有属性组的状态。每个属性组都由此 表中的一行或者其他类型视图表示。属性组的状态反映该属性组的最后一次数据收集尝试或最后一个数据接 收事件的结果。通过检查状态信息,可以确定该代理程序是否正常运行。如果您的代理程序不收集数据,而 是接收数据(事件数据),那么与采样数据相关的属性不会包含有用的数据。只有所列出的前七个属性与事 件数据相关。

"历史"组

此属性组 适合与 Tivoli Data Warehouse 配合使用。

属性描述

以下列表包含关于 Performance Object Status 属性组中每个属性的信息:

"节点"属性:此属性是键属性。

描述

代理程序的受管系统名称。

#### 类型

字符串

仓库名称

NODE

### "时间戳记"属性

#### 描述

收集数据时代理程序的本地时间。

#### 类型

字符串

仓库名称 TIMESTAMP

"查询名称"属性:此属性是键属性。

#### 描述

属性组的名称。

类型

字符串

仓库名称

QUERY\_NAME 或 ATTRGRP

"对象名称"属性

#### 描述

性能对象的名称。

类型

字符串

仓库名称

OBJECT\_NAME 或 OBJNAME

#### "对象类型"属性

#### 描述

性能对象的类型。

#### 类型

具有枚举值的整数。在 Tivoli Enterprise Portal 中将显示字符串。仓库和查询将返回圆括号内显示的值。已定义以下值:

· WMI (0)

- · PERFMON (1)
- · WMI ASSOCIATION GROUP (2)
- · JMX (3)
- · SNMP (4)
- · SHELL COMMAND (5)
- · JOINED GROUPS (6)
- · CIMOM (7)
- $\cdot$  CUSTOM (8)
- · ROLLUP DATA (9)
- · WMI REMOTE DATA (10)
- $\cdot$  LOG FILE (11)

- · JDBC (12)
- · CONFIG DISCOVERY (13)
- $\cdot$  NT EVENT LOG (14)
- · FILTER (15)
- · SNMP EVENT (16)
- · PING (17)
- · DIRECTOR DATA (18)
- · DIRECTOR EVENT (19)
- · SSH REMOTE SHELL COMMAND (20)
- 任何其他值都是 Tivoli Enterprise Portal 中由代理程序返回的值。

#### 仓库名称

OBJECT\_TYPE 或 OBJTYPE

#### "对象状态"属性

#### 描述

性能对象的状态。

#### 类型

具有枚举值的整数。在 Tivoli Enterprise Portal 中将显示字符串。仓库和查询将返回圆括号内显示的值。已定义以下值:

 $\cdot$  ACTIVE (0)

· INACTIVE (1)

任何其他值都是 Tivoli Enterprise Portal 中由代理程序返回的值。

#### 仓库名称

OBJECT\_STATUS 或 OBJSTTS

#### "错误代码"属性

#### 描述

与查询关联的错误代码。

#### 类型

具有枚举值的整数。在 Tivoli Enterprise Portal 中将显示字符串。仓库和查询将返回圆括号内显示的值。已定义以下值:

- · NO ERROR (0)
- · GENERAL ERROR (1)
- · OBJECT NOT FOUND (2)
- · COUNTER NOT FOUND (3)
- · NAMESPACE ERROR (4)
- · OBJECT CURRENTLY UNAVAILABLE (5)
- · COM LIBRARY INIT FAILURE (6)
- · SECURITY INIT FAILURE (7)
- · PROXY SECURITY FAILURE (9)
- · NO INSTANCES RETURNED (10)
- · ASSOCIATOR QUERY FAILED (11)
- · REFERENCE QUERY FAILED (12)
- $\cdot$  NO RESPONSE RECEIVED (13)
- · CANNOT FIND JOINED QUERY (14)

· CANNOT FIND JOIN ATTRIBUTE IN QUERY 1 RESULTS (15)

- · CANNOT FIND JOIN ATTRIBUTE IN QUERY 2 RESULTS (16)
- · QUERY 1 NOT A SINGLETON (17)
- · QUERY 2 NOT A SINGLETON (18)
- NO INSTANCES RETURNED IN QUERY 1 (19)
- NO INSTANCES RETURNED IN QUERY 2 (20)
- · CANNOT FIND ROLLUP QUERY (21)
- · CANNOT FIND ROLLUP ATTRIBUTE (22)
- FILE OFFLINE (23)
- · NO HOSTNAME (24)
- · MISSING LIBRARY (25)
- · ATTRIBUTE COUNT MISMATCH (26)
- · ATTRIBUTE NAME MISMATCH (27)
- · COMMON DATA PROVIDER NOT STARTED (28)
- · CALLBACK REGISTRATION ERROR (29)
- · MDL LOAD ERROR (30)
- · AUTHENTICATION FAILED (31)
- · CANNOT RESOLVE HOST NAME (32)
- · SUBNODE UNAVAILABLE (33)
- · SUBNODE NOT FOUND IN CONFIG (34)
- · ATTRIBUTE ERROR (35)
- · CLASSPATH ERROR (36)
- · CONNECTION FAILURE (37)
- · FILTER SYNTAX ERROR (38)
- FILE NAME MISSING (39)
- · SQL QUERY ERROR (40)
- · SQL FILTER QUERY ERROR (41)
- · SQL DB QUERY ERROR (42)
- · SQL DB FILTER QUERY ERROR (43)
- · PORT OPEN FAILED (44)
- · ACCESS DENIED (45)
- · TIMEOUT (46)
- · NOT IMPLEMENTED (47)
- · REQUESTED A BAD VALUE (48)
- · RESPONSE TOO BIG (49)
- · GENERAL RESPONSE ERROR (50)
- · SCRIPT NONZERO RETURN (51)
- · SCRIPT NOT FOUND (52)
- · SCRIPT LAUNCH ERROR (53)
- · CONF FILE DOES NOT EXIST (54)
- · CONF FILE ACCESS DENIED (55)
- · INVALID CONF FILE (56)
- EIF INITIALIZATION FAILED (57)

- · CANNOT OPEN FORMAT FILE (58)
- · FORMAT FILE SYNTAX ERROR (59)
- · REMOTE HOST UNAVAILABLE (60)
- · EVENT LOG DOES NOT EXIST (61)
- PING FILE DOES NOT EXIST (62)
- · NO PING DEVICE FILES (63)
- · PING DEVICE LIST FILE MISSING (64)
- · SNMP MISSING PASSWORD (65)
- · DISABLED (66)
- · URLS FILE NOT FOUND (67)
- · XML PARSE ERROR (68)
- · NOT INITIALIZED (69)
- · ICMP SOCKETS FAILED (70)

任何其他值都是 Tivoli Enterprise Portal 中由代理程序返回的值。

#### 仓库名称

ERROR\_CODE 或 ERRCODE

### "最近一次收集开始时间"属性

#### 描述

最近一次对该组启动数据收集的时间。

#### 类型

具有枚举值的时间戳记。在 Tivoli Enterprise Portal 中将显示字符串。仓库和查询将返回圆括号 内显示的值。已定义以下值:

· NOT COLLECTED (069123119000000)

· NOT COLLECTED (00000000000000)

任何其他值都是 Tivoli Enterprise Portal 中由代理程序返回的值。

#### 仓库名称

LAST\_COLLECTION\_START 或 COLSTRT

#### "最近一次收集完成时间"属性

#### 描述

最近一次对该组完成数据收集的时间。

#### 类型

具有枚举值的时间戳记。在 Tivoli Enterprise Portal 中将显示字符串。仓库和查询将返回圆括号 内显示的值。已定义以下值:

· NOT COLLECTED (069123119000000)

· NOT COLLECTED (00000000000001)

任何其他值都是 Tivoli Enterprise Portal 中由代理程序返回的值。

#### 仓库名称

LAST\_COLLECTION\_FINISHED 或 COLFINI

#### "最近一次收集持续时间"属性

#### 描述

最近一次对该组完成数据收集的持续时间(秒)。

#### 类型

精确到小数点后2位的实数(32位计数器)

仓库名称

LAST\_COLLECTION\_DURATION 或 COLDURA

#### "平均收集持续时间"属性

#### 描述

该组的所有数据收集的平均持续时间(秒)。

#### 类型

具有枚举值且精确到小数点后 2 位的实数(32 位计数器)。在 Tivoli Enterprise Portal 中将显示 字符串。仓库和查询将返回圆括号内显示的值。已定义以下值:

•NO DATA (-100)

任何其他值都是 Tivoli Enterprise Portal 中由代理程序返回的值。

#### 仓库名称

AVERAGE\_COLLECTION\_DURATION 或 COLAVGD

#### "刷新时间间隔"属性

#### 描述

对该组进行刷新的时间间隔(秒)。

#### 类型

整数(32位计数器)

仓库名称

REFRESH\_INTERVAL 或 REFRINT

#### "收集次数"属性

#### 描述

自代理程序启动以来对该组进行收集的次数。

#### 类型

整数(32位计数器)

#### 仓库名称

NUMBER\_OF\_COLLECTIONS 或 NUMCOLL

#### "高速缓存命中次数"属性

#### 描述

通过高速缓存满足该组外部数据请求的次数。

#### 类型

整数(32位计数器)

#### 仓库名称

CACHE\_HITS 或 CACHEHT

#### "高速缓存未命中次数"属性

#### 描述

未能通过高速缓存满足该组外部数据请求的次数。

#### 类型

整数(32 位计数器)

### 仓库名称

CACHE\_MISSES 或 CACHEMS

#### "高速缓存命中百分比"属性

描述

通过高速缓存满足该组外部数据请求的百分比。

类型

精确到小数点后2位的实数(32位计数器)

仓库名称

CACHE\_HIT\_PERCENT 或 CACHPCT

"跳过的时间间隔数"属性

#### 描述

由于应该开始下一次收集时上一次收集仍在运行而跳过后台数据收集的次数。

类型

整数(32位计数器)

仓库名称

INTERVALS\_SKIPPED 或 INTSKIP

# Thread Pool Status 属性组

Thread Pool Status 属性组包含反映内部线程池(用于以异步方式收集数据)状态的信息。 以下是构成此属性组的属性列表。粗体名称显示了属性在 Tivoli Enterprise Portal 中的显示方式。 以下列表包含关于 Thread Pool Status 属性组中每个属性的信息:

"节点"属性-此属性是键属性

```
描述
```

代理程序的受管系统名称

类型

字符串

名称

属性名称 Node

列名

ORIGINNODE

"时间戳记"属性

#### 描述

当构建数据行并将其从代理程序发送到 Tivoli Enterprise Monitoring Server 时,从代理程序系统收集 到的时间。或者进行存储以记录历史数据。它表示代理程序系统的本地时区。

### 类型

时间

### 名称

属性名称

Timestamp

列名

TIMESTAMP

"线程池大小"属性

描述

线程池中当前存在的线程数。

类型

整数

### 名称

属性名称

Thread\_Pool\_Size

列名

THPSIZE

"线程池最大大小"属性

### 描述

线程池中允许存在的最大线程数。 **类型** 整数

### 名称

属性名称 Thread\_Pool\_Max\_Size

列名

# "线程池活动线程数"属性

### 描述

线程池中当前正在执行工作的线程数。

### 类型

整数

### 名称

属性名称 Thread\_Pool\_Active\_Threads 列名 TPACTTH

### "线程池平均活动线程数"属性

**描述** 线程池中同时在执行工作的平均线程数。

# 类型

整数

# 名称

属性名称 Thread\_Pool\_Avg\_Active\_Threads 列名

ti TPAVGAT

# "线程池最小活动线程数"属性

# 描述

线程池中同时在执行工作的最小线程数。 **类型** 整数 名称

属性名称

Thread\_Pool\_Min\_Active\_Threads

列名

TPMINAT

"线程池最大活动线程数"属性

#### 描述

线程池中同时在执行工作的最大线程数。

### 类型

整数

### 名称

属性名称 Thread\_Pool\_Max\_Active\_Threads

列名

TPMAXAT

### "线程池队列长度"属性

#### 描述

线程池队列中当前正在等待的作业数。 **类型** 

整数

#### 名称

属性名称 Thread\_Pool\_Queue\_Length 列名

TPQLGTH

"线程池平均队列长度"属性

### 描述

此次运行期间线程池队列的平均长度。

#### 类型

整数

### 名称

属性名称

Thread\_Pool\_Avg\_Queue\_Length

列名

TPAVGQL

### "线程池最小队列长度"属性

### 描述

线程池队列达到的最小长度。

### 类型

整数

### 名称

属性名称 Thread\_Pool\_Min\_Queue\_Length 列名

TPMINQL

#### "线程池最大队列长度"属性

#### 描述

线程池队列达到的最大长度。

#### 类型

整数

### 名称

属性名称 Thread\_Pool\_Max\_Queue\_Length 列名

TPMAXQL

#### "线程池平均作业等待时间"属性

#### 描述

作业等待线程池队列所花费的平均时间。

#### 类型

整数

### 名称

属性名称 Thread\_Pool\_Avg\_Job\_Wait 列名

**TPAVJBW** 

#### "线程池作业总数"属性

#### 描述

自代理程序启动以来池中所有线程完成的作业数。

#### 类型

整数

### 名称

属性名称

Thread\_Pool\_Total\_Jobs

#### 列名

TPTJOBS

# "事件日志"属性节点

"事件日志"属性组包含所有与应用程序相关的最近事件日志条目。

缺省情况下,代理程序只显示在代理程序启动后发生的事件。事件发生1小时后,将从"事件日志"视图中移除。

以下列表包含关于"事件日志"属性组中每个属性的信息:

"节点"属性-此属性是键属性

**描述** 代理程序的受管系统名称 **类型** 字符串 名称

属性名称

Node

列名

ORIGINNODE

### "日志名称"属性

### 描述

事件日志 - 应用程序日志、系统日志、安全日志或特定于应用程序的日志

### 类型

字符串

### 名称

属性名称 Log\_Name

# 列名

LOGNAME

### "事件源"属性

### 描述

应用程序所定义的事件源

### 类型

字符串

### 名称

属性名称 Event\_Source

# \_\_\_\_\_\_\_\_\_\_列名

### "事件类型"属性

### 描述

事件类型 - Error(0)、Warning(1)、Informational(2)、Audit\_Success(3)、Audit\_Failure(4) 和 Unknown(5)

### 类型

整数

# 名称

属性名称 Event\_Type

列名

EVTTYPE

### "事件标识"属性

# 描述

事件的标识 类型 整数 名称 属性名称 Event\_ID

列名 **EVTID** "事件类别"属性 描述 事件的类别 类型 字符串 名称 属性名称 Event\_Category 列名 **EVTCATEG** "消息"属性 描述 事件消息 类型 字符串 名称 属性名称 Message 列名 MESSAGE "生成时间"属性 描述 生成事件的时间 类型 时间 名称 属性名称 Time\_Generated 列名 TIMESTAMP

# 日志文件摘要

在数据源的高级属性中选择对应选项后,此属性组的特性将包含在摘要属性组中。

如果在数据源的高级属性中选择了将特性包含在摘要属性组中,那么将针对每个"日志文件"数据源创建一个"摘要"节点。摘要节点的名称为末尾添加 Summary 的数据源名称。

以下列表包含关于"日志文件摘要"属性组中每个缺省特性的信息。这些特性始终包含在摘要属性组中。如 果选中**将特性包含在摘要属性组中**(请参阅<u>第88页的『监视日志文件』</u>中的步骤<u>第92页的『9』</u>),那么 该日志属性组的摘要属性组还将包含您选择的每个特性。值为日志文件属性组中对应特性的副本。

所有添加的特性共同组成一个键,每一组唯一键在摘要表中都有对应的一行。此行指示在时间间隔内接收到 的日志记录数(在这些记录中,提供的所有键都与相应特性中报告的值匹配)。

"节点"特性 - 此特性是键特性

描述 代理程序的受管系统名称 类型 字符串 名称 特性名称 Node 列名 ORIGINNODE "时间戳记"特性

#### 描述

收集数据时代理程序的本地时间

类型

# 时间

名称

特性名称 Timestamp

列名

TIMESTAMP

#### "时间间隔单位"特性

描述

两次生成摘要特性之间间隔的秒数

#### 类型

整数 (标尺)

名称

特性名称 \_Interval\_Unit 列名 IU

"时间间隔"特性

#### 描述

当前时间间隔在下一个更大时间单位内的偏移量(例如,在1小时中的分钟数) **类型** 整数(标尺) **名称 特性名称** \_Interval

\_\_\_\_\_

列名 INV

### "发生次数"特性

描述

在时间间隔内记录的发生次数 **类型** 整数(标尺) 名称 特性名称 \_Occurrences 列名 OCC

"本地时间戳记"特性

#### 描述

生成摘要数据的时间

#### 类型

时间戳记

#### 名称

特性名称

\_LocalTimeStamp

列名

LTS

"日期时间"特性

#### 描述

生成摘要数据的时间

类型

字符串

### 名称

特性名称 \_Date\_Time

\_ 列名

DT

"时间间隔单位名称"特性

#### 描述

时间间隔单位的文字描述

#### 类型

字符串

#### 名称

```
特性名称
__Interval_Unit_Name
列名
IUN
```

# "AIX 二进制日志"属性组

"AIX 二进制日志"属性组显示来自AIX 二进制日志的事件,这些事件由提供的 errpt 命令字符串选择。

以下列表包含关于"AIX二进制日志"属性组中每个属性的信息:

注: Agent Builder 不允许移除 Identifier、ErrptTimestamp、Type、Class、ResourceName 和 Description 属性,以及对它们重新排序或更改大小。代理程序将根据文本行中的列解析 errpt 命令所返 回的数据。这些列由 Identifier、ErrptTimestamp、Type、Class、ResourceName 及 Description 特性的顺序及大小定义。移除这些特性、将它们重新排序或更改它们的大小将更改各列所对 应的特性。这将导致 Tivoli Monitoring 中显示的结果行不正确。

但是, 您可将这些特性重命名。

"节点"特性 - 此特性是键特性

描述

代理程序的受管系统名称

**类型** 字符串

名称

特性名称 Node

列名

#### Identifier 属性 - 此属性是键属性

描述

errpt 所报告的事件标识

类型

字符串

名称

特性名称 Identifier

列名

IDENTIFIER

### ErrptTimestamp 特性

#### 描述

errpt 所报告的事件记录时间。

**注:** 此特性在运行时处于隐藏状态。此特性包含原始值。从此特性派生的其他特性以更易使用的形式显示值。为此,此属性可以在 Agent Builder 中使用,而在 Tivoli Monitoring 环境中,运行时将缺省处于不可见状态。如果您希望此特性可见,请在 Agent Editor 中的"数据源定义"页面中选中此特性,然后选中在 Tivoli Enterprise Portal 中显示特性。

类型

字符串

名称

特性名称

ErrptTimestamp

列名

ERRPTTIMES

### 类型

描述

errpt 所报告的单字符事件类型,即,下列其中一项: I(NFO)、P(END/ERF/ERM)、T(EMP)和U(NKN)

类型

字符串

名称

特性名称 Type 列名

TYPE

#### "类"特性-此特性是键特性

#### 描述

errpt 所报告的事件类,即,下列其中一项: Hardware、Software、Operator 和 Undertermined。这些是枚举值。与情境配合使用的原始值为 H、S、O 和 U。

#### 类型

字符串

### 名称

特性名称 Class 列名 CLASS

#### ResourceName

#### 描述

errpt 所报告的资源名称,用于标识错误记录的来源

#### 类型

字符串

#### 名称

特性名称 ResourceName

#### 列名

RESOURCENA

#### "描述"特性

#### 描述

errpt 所报告的描述,这通常是描述错误性质的简短文本消息

#### 类型

字符串

名称

特性名称

描述

列名

DESCRIPTIO

#### LogFile 特性

#### 描述

二进制 errpt 日志的全名(包括路径)。

注:此特性在运行时处于隐藏状态。此特性包含原始值。从此特性派生的其他特性以更易使用的形式显示值。为此,此属性可以在 Agent Builder 中使用,而在 Tivoli Monitoring 环境中,运行时将缺省处于不可见状态。如果您希望此特性可见,请在 Agent Editor 中的"数据源定义"页面中选中此特性,然后选中在 Tivoli Enterprise Portal 中显示特性。

#### 类型

字符串

名称

特性名称 LogFile 列名 LOGFILE

### "系统"特性

#### 描述

在其中收集错误的系统的主机名

### 类型

字符串

### 名称

特性名称 System

列名

SYSTEM

### LogName 特性

#### 描述

从中收集记录的二进制 errpt 日志的基本名称

### 类型

字符串

#### 名称

特性名称

LogName

### 列名

LOGNAME

#### LogPath 特性

#### 描述

从中收集记录的二进制 errpt 日志所在目录的名称

### 类型

字符串

### 名称

特性名称 LogPath

列名 LOGPATH

# EntryTime 特性

### 描述

errpt 所报告的事件记录时间(采用 Tivoli 时间戳记格式)。此时间不一定与代理程序接收事件的时间(记录在 **Timestamp** 字段中)完全相同。

#### 类型

时间戳记

### 名称

特性名称 EntryTime 列名 ENTRYTIME

# "监视器和通知"属性组

"监视器和通知"属性组的定义。

前4项特定于监视器,最后1项特定于通知(全部都与 JMX 相关)。

列出的每一项都带有指示,指出其是否基于事件。对于不基于事件的属性组,将在需要时收集数据。对于基于事件的属性组,代理程序将保留最近收到的100个事件的高速缓存。这些事件用于响应来自Tivoli Enterprise Portal 的请求。这些事件将立即转发,以用于按情境分析和进行仓储。

#### 计数器通知

"计数器通知"属性组是不基于事件的属性组,用于发送所有计数器监视器所接收的事件。

以下列表包含关于"计数器通知"属性组中每个属性的信息:

#### "节点"属性 - 此属性是键属性

描述

代理程序的受管系统名称

类型

字符串

名称

属性名称

Node

列名 ORIGINNODE

#### "时间戳记"属性

#### 描述

收集数据时代理程序的本地时间

类型

时间

名称

属性名称

Timestamp

列名

TIMESTAMP

"通知类型"属性

#### 描述

接收到的通知类型。描述 MBean 的受观测属性如何触发通知。

类型

字符串

#### 名称

属性名称 Notification\_Type 列名 NOTIFICATI

#### "监视器标识"属性

描述 生成此通知的监视器的监视器标识 类型 整数 名称 属性名称 Monitor\_ID 列名 MONITOR\_ID "受观测 MBean"属性 描述 正在监视其属性的 MBean 类型 字符串 名称 属性名称 Observed\_MBean 列名 OBSERVED\_M "受观测属性"属性 描述 "受观测 MBean"中监视的属性的名称 类型 字符串 名称 属性名称 Observed\_Attribute 列名 OBSERVED\_A "阈值"属性 描述 监视器的当前阈值

### 类型

字符串

### 名称

属性名称 Threshold

列名

THRESHOLD

### "偏移量"属性

描述

此属性每次超出阈值时, 向阈值增加的值。此值构成新的阈值。

类型 字符串 名称 属性名称 Offset 列名 OFFSET "模数"属性 描述 此属性的最大值。达到此值时,此属性将回滚并从0开始重新计数。 类型 整数 名称 属性名称 Modulus 列名 MODULUS "计数器值"属性 描述 触发通知的计数器值 类型 整数 名称 属性名称 Counter\_Value 列名 COUNTER\_VA "通知时间戳记"属性 描述 触发通知的时间 类型 时间 名称 属性名称 Notification\_Time\_Stamp 列名 **NOTIFICATO** "通知消息"属性 描述 通知中的消息 类型

字符串

名称 属性名称 Notification\_Message 列名 NOTIFICAT1

### 标尺通知

"标尺通知"属性组是并非基于事件的属性组,用于发送所有标尺监视器所接收的事件。 以下列表包含关于"标尺通知"属性组中每个属性的信息:

"节点"属性 - 此属性是键属性

**描述** 代理程序的受管系统名称

类型

字符串

名称

属性名称

Node 列名

ORIGINNODE

"时间戳记"属性

描述

收集数据时代理程序的本地时间 **类型** 

时间

名称

属性名称 Timestamp 列名

⊐ TIMESTAMP

"通知类型"属性

#### 描述

接收到的通知类型。描述 MBean 的受观测属性如何触发通知。 **类型** 字符串

# 名称

属性名称 Notification\_Type

列名

NOTIFICATI

"监视器标识"属性

### 描述

生成此通知的监视器的监视器标识

### 类型

整数

名称 属性名称 Monitor\_ID 列名 MONITOR\_ID

"受观测 MBean"属性

#### 描述

正在监视其属性的 MBean

#### 类型

字符串

### 名称

属性名称

Observed\_MBean

## 列名

OBSERVED\_M

### "受观测属性"属性

#### 描述

~"受观测 MBean"中监视的属性的名称

#### 类型

字符串

#### 名称

属性名称 Observed\_Attribute

#### 列名

OBSERVED\_A

### "阈值下限"属性

#### 描述

受监视器监视以待受观测属性跨越的阈值

#### 类型

字符串

### 名称

属性名称 Low\_Threshold 列名

LOW\_THRESH

#### "阈值上限"属性

#### 描述

受监视器监视以待受观测属性跨越的阈值 **类型** 字符串 名称

属性名称 High\_Threshold

## 列名 HIGH\_THRES

# "标尺值"属性

#### 描述

触发通知的标尺值

### 类型

字符串

### 名称

属性名称 Gauge\_Value

### 列名

MODULUSGAUGE\_VALU

### "通知时间戳记"属性

描述

触发通知的时间

#### 类型

时间

## 名称

属性名称 Notification\_Time\_Stamp 列名

NOTIFICAT0

## "通知消息"属性

### 描述

通知中的消息

#### 类型

字符串

### 名称

属性名称

Notification\_Message

### 列名

NOTIFICAT1

### 已注册的监视器

"已注册的监视器"属性组是基于事件的属性组,它显示由代理程序创建的所有 JMX 监视器的列表。 以下列表包含关于"已注册的监视器"属性组中每个属性的信息:

### "节点"属性-此属性是键属性

#### 描述

代理程序的受管系统名称 **类型** String

### 名称

特性名称 Node

列名称 ORIGINNODE "时间戳记"属性 描述 收集数据时代理程序的本地时间 类型 时间 名称 特性名称 Timestamp 列名称 TIMESTAMP "监视器标识"属性 - 此属性是键属性 描述 监视器的唯一整数标识 类型 整数 名称 特性名称 Monitor\_ID 列名称 MONITOR\_ID "监视器参数"属性 描述 这是用于创建监视器的参数 类型 String 名称 特性名称 Monitor\_Parameters 列名称 MONITOR\_PA "监视器名称"属性 描述 监视器 MBean 的 JMX 对象名 类型 String 名称 特性名称 Monitor\_Name 列名称 MONITOR\_NA

### 字符串通知

"字符串通知"属性组是不基于事件的属性组,用于发送所有字符串监视器所接收的事件。 以下列表包含关于"字符串通知"属性组中每个属性的信息:

"节点"属性 - 此属性是键属性

```
描述
```

代理程序的受管系统名称

类型

字符串

名称

属性名称 Node

列名

ORIGINNODE

### "时间戳记"属性

#### 描述

收集数据时代理程序的本地时间

### 类型

时间

### 名称

属性名称 Timestamp

列名 TIMESTAMP

### "通知类型"属性

#### 描述

接收到的通知类型。描述 MBean 的受观测属性如何触发通知。 类型 字符串 名称

#### 百仰

属性名称

Notification\_Type

列名

NOTIFICATI

"监视器标识"属性 - 此属性是键属性

### 描述

监视器的唯一整数标识

### 类型

整数

### 名称

属性名称 Monitor\_ID 列名

MONITOR\_ID

#### "受观测 MBean"属性

### 描述

正在监视其属性的 MBean

# 类型

字符串

# 名称

属性名称 Observed\_MBean

列名

OBSERVED\_M

### "受观测属性"属性

### 描述

"受观测 MBean"中监视的属性的名称 类型 字符串 名称

属性名称 Observed\_Attribute 列名 OBSERVED\_A

### "比较字符串"属性

描述

比较操作中使用的字符串 类型 字符串

# 名称

属性名称 Compare\_String

# 列名

COMPARE\_ST

### "字符串值"属性

### 描述

触发通知的属性值

### 类型

字符串

### 名称

属性名称 String\_Value

列名

STRING\_VAL

### "通知时间戳记"属性

### 描述

触发通知的时间

类型

时间

### 名称

属性名称 Notification\_Time\_Stamp 列名

NOTIFICAT0

### "通知消息"属性

### 描述

通知中的消息

类型

字符串

名称

属性名称

Notification\_Message

#### 列名

NOTIFICAT1

# "SNMP 事件"属性组

"SNMP事件"属性组用于接收陷阱和通知。这些属性组是基于事件的属性组

以下列表包含关于"SNMP事件"属性组中每个属性的信息:

注:您可以更改这些属性的缺省显示名称。这些显示名称与各个属性的内部标识不同。

### Enterprise\_OID

生成陷阱的企业 OID。

#### Source\_Address

发送陷阱的 SNMP 代理程序的主机名或 IP 地址。

#### Generic\_Trap

从接收到的陷阱中抽取的通用陷阱号。可能的值包括:

- 0 ColdStart
- 1 WarmStart
- 2 LinkDown
- 3 LinkUp
- 4 Authentication Failure
- 5 EGPNeighborLoss

#### Specific\_Trap

从接收到的陷阱中抽取的特定于企业的陷阱号。仅当 Generic\_Trap = 6 时适用。

#### Alert\_Name

陷阱配置文件的定义中指定的陷阱名称。

#### Category

陷阱配置文件的定义中指定的陷阱类别。

#### 描述

陷阱配置文件的定义中指定的陷阱描述。最长描述长度为 256 个字符。

#### Enterprise\_Name

在陷阱配置文件中指定,并通过陷阱对象标识确定的陷阱企业名称。

#### Source\_Status

生成陷阱的代理程序在发送陷阱后所处的状态(在陷阱配置文件中的陷阱定义中指定)。

#### Source\_Type

生成陷阱的代理程序的类型(在陷阱配置文件的陷阱定义中指定)。

#### **Event\_Variables**

在陷阱协议数据单元 (PDU) 中接收到的变量绑定 (VarBind) 数据。此字符串的构造如下:

{OID[type]=value}{OID[type]=value}{oid[type]=value}...

其中:

#### oid

MIB 变量对象标识

#### type

SMI 数据类型

#### value

变量值

{}

每个三元组都括在花括号({})内。

注: 属性 Alert Name、Category、Description、Enterprise\_Name、Source\_Status 和 Source\_Type 提供更多信息。在"SNMP MIB 浏览器"窗口中,选中包括用于显示陷阱配置文件所定义信息的属性复选框即可包括这些属性。

# "JMX 事件"属性组

"JMX 事件"属性组用于接收来自 MBean Server 的通知。

这些属性组是不基于事件的属性组,使用以下可由代理程序开发者编辑的属性生成。

以下列表包含关于"JMX事件"属性组中每个属性的信息:

"节点"属性 - 此属性是键属性

#### 描述

代理程序的受管系统名称

**类型** 字符串

名称

属性名称

Node

列名

ORIGINNODE

"时间戳记"属性

描述 收集数据时代理程序的本地时间 类型 时间 名称 属性名称 Timestamp 列名

TIMESTAMP

#### "类型"属性

描述 通知类型 类型 字符串 名称 属性名称 Туре 列名 TYPE "源"属性 描述 这是导致发送通知的 MBean 类型 字符串 名称 属性名称 Source 列名 SOURCE "序号"属性 描述 通知对象中的序号 类型 字符串 名称 属性名称 Sequence\_Number 列名 SEQUENCE\_N "消息"属性 描述 通知消息 类型 字符串 名称 属性名称 Message 列名 MESSAGE "用户数据"属性 描述 通知中的用户数据对象

### 类型

字符串

名称 属性名称 User\_Data 列名 USER\_DATA

# Ping 属性组

Ping 属性组包含发送到多列设备的 ICMP Ping 的结果。

以下列表包含关于 Ping 属性组中每个属性的信息:

"节点"属性-此属性是键属性

#### 描述

代理程序的受管系统名称。

类型

字符串

名称

属性名称 Node

列名

ORIGINNODE

"时间戳记"属性

#### 描述

当构建数据行并将其从代理程序发送到 Tivoli Enterprise Monitoring Server 时,从代理程序系统收集 到的时间。或者进行存储以记录历史数据。它表示代理程序系统的本地时区。

#### 类型

时间

### 名称

属性名称

Timestamp

列名

TIMESTAMP

"地址"属性-此属性是键属性

#### 描述

受监视主机的 IP 地址。

#### 类型

具有枚举值的字符串。如果 IP 地址未知,那么会显示值 UNKNOWN\_ADDRESS。对于此枚举,仓库和查询将返回 0.0.0.0。任何其他 IP 地址值均按原样显示。

名称

属性名称

Address

列名

PNGADDR

"设备条目"属性 - 此属性是键属性

描述

此节点在设备列表文件中的条目。

类型

字符串

名称

属性名称

Device\_Entry

列名

PINGDEVC

"当前响应时间"属性

### 描述

受管节点的 ICMP 请求的当前网络响应时间(毫秒)。

### 类型

具有枚举值的整数。在 Tivoli Enterprise Portal 中将显示字符串。仓库和查询会返回数字。已定义的 值为 TIMEOUT (-1) 和 SEND\_FAILURE (-2)。所有其他值均显示数字值。

### 名称

属性名称 Current\_Response\_Time

列名

PINGRSTM

### "名称"属性

### 描述

受管节点的主机名。如果无法通过 DNS 解析节点地址,那么将显示点分十进制 IP 地址。

#### 类型

具有枚举值的字符串。如果主机名未知,那么会显示值 UNKNOWN\_HOSTNAME。对于此枚举,仓 库和查询将返回 0.0.0.。任何其他主机名值均按原样显示。

### 名称

属性名称

Name

列名

PNGNAME

#### "节点描述"属性

### 描述

受管节点的描述。

类型

字符串

### 名称

属性名称

Node\_Description

列名

PNGDESC

"节点状态"属性

### 描述

受管节点的当前运行状态。

#### 类型

具有枚举值的整数。在 Tivoli Enterprise Portal 中将显示字符串。仓库和查询会返回数字。已定义的 值为 INVALID(-2)、UNKNOWN(-1)、INACTIVE(0) 和 ACTIVE(1)。

#### 名称

属性名称

Node\_Status

列名

PNGSTAT

### "节点类型"属性

#### 描述

受管节点的类型。如果节点处于联机状态,那么类型为 IP Node。如果节点处于脱机状态,那么类型为 Unknown。

#### 类型

具有枚举值的整数。在 Tivoli Enterprise Portal 中将显示字符串。仓库和查询会返回数字。已定义的 值为 UNKNOWN(0) 和 IP NODE(1)。

#### 名称

属性名称

Node\_Type

列名 PNGTYPE

#### 状态时间戳记

**描述** 最近一次检查节点的日期和时间。 **类型** 时间

名称

属性名称 Status\_Timestamp 列名

PNGTMSP

### HTTP 属性组

受管 URL 和 URL 对象这两个 HTTP 属性组用于接收来自 URL 及其中对象的信息。

有关 "受管 URL" 和 "URL 对象" 表中使用的语法的信息, 请参阅 第 119 页的 『HTTP 属性的特定字段』。

#### 受管 URL

以下列表包含关于"受管 URL"属性组中每个属性的信息:

"节点"属性-此属性是键属性

#### 描述

代理程序的受管系统名称 **类型** 字符串 名称 属性名称 Node 列名 ORIGINNODE

#### "时间戳记"属性

描述

收集数据时代理程序的本地时间

类型

时间

名称

属性名称

Timestamp

列名\_

TIMESTAMP

URL 属性 - 此属性是键属性

#### 描述

受监视的 URL。

类型、

字符串

名称

属性名称

URL

列名

HTTPURL

### "响应时间"属性

#### 描述

下载响应所用的时间(毫秒)。

#### 类型

具有枚举值的整数。在 Tivoli Enterprise Portal 中显示字符串,仓库和查询返回数字。已定义的值为 TIMEOUT (-1)。

### 名称

属性名称 Response\_Time

列名

HTTPURL

"页面大小"属性

#### 描述

HTTP 请求所返回页面的大小。

类型

具有枚举值的整数。在 Tivoli Enterprise Portal 中显示字符串,仓库和查询返回数字。已定义的值为 NO\_RESPONSE\_RECEIVED(-1)。

名称

属性名称 Page\_Size 列名

PAGESZ

"页面对象数"属性

#### 描述

与受监视页面相关联的对象总数。

#### 类型

具有枚举值的整数。在 Tivoli Enterprise Portal 中显示字符串,仓库和查询返回数字。已定义的值为 NOT\_COLLECTED(-1)。

名称

#### 属性名称

Page\_Objects

列名

PGOBJS

"对象总大小"属性

#### 描述

HTTP 请求所返回页面的大小。

#### 类型

具有枚举值的整数。在 Tivoli Enterprise Portal 中显示字符串,仓库和查询返回数字。已定义的值为 NOT\_COLLECTED(-1)。

#### 名称

属性名称 Total\_Object\_Size 列名 TOTOSZ

"页面标题"属性

#### 描述

所接收 URL 页面的页面标题。

#### 类型

字符串

#### 名称

属性名称

Page\_Title

列名

PAGETTL

### "服务器类型"属性

描述

在目标 URL Web 站点上使用的服务器的类型。

## 类型

字符串
名称

属性名称 Server\_Type 列名 SRVTYP

"响应代码"属性

描述

HTTP 请求的响应代码。

#### 类型

具有枚举值的整数。在 Tivoli Enterprise Portal 中显示字符串,仓库和查询返回数字。已定义的值为 NO\_RESPONSE\_RECEIVED(-1)。

名称

属性名称

Response\_Code

列名

CODE

"状态"属性

#### 描述

当前受管 URL 的状态(OK 或状态描述)。

类型

字符串

名称

属性名称 Status 列名 STATUS

"URL 别名"属性

```
描述
```

URL 的用户指定别名。

## 类型

字符串

## 名称

属性名称 URL\_Alias

列名

ALIAS

"用户数据"属性

描述

随 URL 一起指定的用户数据。 **类型** 字符串 **名称 属性名称** User\_Data 列名 USER

#### URL 对象

以下列表包含关于"URL 对象"属性组中每个属性的信息: "**节点"属性-此属性是键属性** 

#### 描述

代理程序的受管系统名称

## 类型

字符串

名称

属性名称 Node

列名

ORIGINNODE

## "时间戳记"属性

#### 描述

收集数据时代理程序的本地时间 **类型** 时间

## 名称

属性名称 Timestamp

## 列名

TIMESTAMP

URL 属性 - 此属性是键属性

#### 描述

受监视的 URL。

## 类型

字符串

## 名称

属性名称 URL 列名 HTTPURL

#### 5. 6. 46. 11 H

"对象名称"属性

描述
 目标 URL 内的页面对象的名称。
 类型
 字符串
 名称

#### 属性名称 Object\_Name

列名

ONAME

#### "对象大小"属性

#### 描述

目标 URL 内的页面对象的大小(字节)。

#### 类型

具有枚举值的整数。在 Tivoli Enterprise Portal 中将显示字符串。仓库和查询会返回数字。已定义的 值为 NOT\_COLLECTED (-1) 和 OBJECT\_NOT\_FOUND (-2)。所有其他值均显示数字值。

## 名称

属性名称

Object\_Size

列名

SIZE

"对象响应时间"属性

## 描述

下载对象所用的时间(毫秒)。

## 类型

具有枚举值的整数。在 Tivoli Enterprise Portal 中将显示字符串。仓库和查询会返回数字。已定义的 值为 NOT\_COLLECTED (-1)、NO\_RESPONSE\_RECEIVED (-2)和 STATUS\_CODE\_ERROR (-3)。所有其他值均显示数字值。

## 名称

属性名称 Object\_Response\_Time 列名

ORTIME

# 发现属性组

此属性组代表针对子节点类型定义的子节点实例集合

创建子节点类型时,将创建一个属性组,它代表针对该子节点类型定义的子节点实例集合。其中每个属性组 包含相同的属性集合。

以下列表包含关于"发现"属性组中的每个属性的信息。粗体字名称显示属性在 Tivoli Enterprise Portal 中的显示方式:

## "节点"属性 - 此属性是键属性

描述

代理程序的受管系统名称 **类型** 字符串 名称 属性名称 Node 列名称

ORIGINNODE

"时间戳记"属性

描述

构建数据行并将其发送到 Tivoli Enterprise Monitoring Server(或进行存储以用于历史记录用途)时,从代理程序系统收集的时间。它表示代理程序系统的本地时区。

类型

时间

名称

属性名称

Timestamp

列名称

TIMESTAMP

"子节点 MSN"属性

描述

子节点代理程序的"受管系统名称"。

类型

字符串

## 名称

属性名称 Subnode\_MSN 列名称 SN\_MSN

"子节点亲缘关系"属性

描述

子节点代理程序的亲缘关系。 **类型** 字符串

名称

属性名称 Subnode\_Affinity 列名称 SN\_AFFIN

"子节点类型"属性

描述

此子节点的节点类型。

类型

字符串

#### 名称

属性名称 Subnode\_Type 列名称 SN\_TYPE

"子节点资源名称"属性

#### 描述

子节点代理程序的资源名称。

类型 字符串 名称 属性名称 Subnode\_Resource\_Name 列名称 SN\_RES "子节点版本"属性 描述 子节点代理程序的版本。 类型 名称 属性名称 Subnode\_Version 列名称 SN\_VER

# Take Action Status 属性组

Take Action Status 属性组包含代理程序已处理的操作的状态。

此属性组基于事件,并包含关于 Take Action Status 属性组中每个属性的信息:

"节点"属性 - 此属性是键属性

## 描述

代理程序的受管系统名称。

## 类型

字符串

# 名称

属性名称 Node

列名

ORIGINNODE

## "时间戳记"属性

## 描述

这是构建数据行并将其从代理程序发送到 Tivoli Enterprise Monitoring Server 时,或进行存储以记录历史数据时,从代理程序系统收集的时间。它表示代理程序系统的本地时区。

#### 类型

时间

## 名称

属性名称

Timestamp

## 列名

TIMESTAMP

## "操作名称"属性

描述

已运行的操作的名称

类型

字符串

#### 名称

属性名称

Action\_Name

列名

TSKNAME

## "操作状态"属性

#### 描述

操作的状态。

#### 类型

具有枚举值的整数。值包括: 正常 (0)、不适用 (1)、一般错误 (2)、警告 (3)、不在运行中 (4)、依赖 项不在运行中 (5)、已在运行中 (6)、必备项不在运行中 (7)、已超时 (8)、不存在 (9)、未知 (10)、依 赖项仍在运行中 (11)、用户权限不足 (12)

## 名称

#### 属性名称

Action\_Status

列名

TSKSTAT

## "操作应用程序返回码"属性

描述

操作所启动的应用程序的返回码。

#### 类型

整数

#### 名称

属性名称

Action\_App\_Return\_Code

列名

## TSKAPRC

"操作消息"属性

## 描述

与操作返回码关联的消息。

## 类型

字符串

#### 名称

属性名称

Action\_Message

#### 列名 TSKMSGE

# "操作实例"属性

## 描述

与运行操作所生成的输出关联的实例。如果操作是系统命令,那么实例是命令输出的行号。

#### 类型

字符串

## 名称

属性名称 Action\_Instance

列名 TSKINST "操作结果"属性 描述 运行操作所生成的输出。 类型 字符串 名称 属性名称 Action\_Results 列名 TSKOUTP "操作命令"属性 描述 操作所运行的命令。 类型 字符串 名称 属性名称 Action\_Command 列名 TSKCMND "操作节点"属性 描述 运行操作的节点。 类型 字符串 名称 属性名称 Action\_Node 列名 TSKORGN "操作子节点"属性 描述 运行操作的子节点。 类型 字符串 名称 属性名称 Action\_Subnode 列名 TSKSBND "操作标识"属性 描述 操作的标识。 类型

整数

名称

属性名称

Action\_ID

列名 TSKID

"操作类型"属性

描述

操作的类型。

类型

具有枚举值的整数。在 Tivoli Enterprise Portal 中将会显示字符串,仓库和查询将返回数字。已定义的值为:未知 (0)、自动化 (1)。

名称

属性名称

Action\_Type

列名

TSKTYPE

"操作所有者"属性

描述

启动操作的情境或用户的名称。

类型

字符串

名称

属性名称

Action\_Owner

列名

TSKOWNR

# "日志文件状态"属性组

"日志文件状态"属性组包含的信息反映此代理程序所监视的日志文件的状态。

如果您具有日志属性组并且代理程序处于缺省最低的 Tivoli Monitoring V6.2.1 或更高版本,那么包含"日志 文件状态"属性组。"日志文件状态"属性组包含两个定义为 64 位数字的属性,以便能够处理大文件。64 位数字属性支持由 Tivoli Monitoring V6.2.1 或更高版本提供。

以下列表包含关于"日志文件状态"属性组中的每个属性的信息:

"节点"属性-此属性是键属性

描述

代理程序的受管系统名称。

类型

字符串

名称

属性名称

节点

列名 ORIGINNODE

#### "时间戳记"属性

```
描述
```

此值是构建数据行并从代理程序发送到 Tivoli Enterprise Monitoring Server 时,从代理程序系统进行 收集时的时间。或进行存储以记录历史数据。它表示代理程序系统的本地时区。

类型

时间

## 名称

属性名称

时间戳记

列名

TIMESTAMP

"表名"属性 - 此属性是键属性

描述

正在其中监视此日志的表的名称

类型

字符串

### 名称

属性名称

表名

列名 TBLNAME

"文件名"属性 - 此属性是键属性

描述

正被监视的文件的名称

#### 类型

字符串

#### 名称

属性名称

文件名

列名

FILNAME

## "正则表达式模式"属性-此属性是键属性

#### 描述

这是导致此文件被监视的正则表达式模式(如果有)

## 类型

字符串

## 名称

**属性名称** 正则表达式模式

列名 REPATRN

## "文件类型"属性

描述

此文件的类型(常规文件或管道)

#### 类型

具有枚举值的整数。在 Tivoli Enterprise Portal 中将显示字符串。已定义的值为 UNKNOWN(0)、 REGULAR FILE(1) 和 PIPE(2) 名称

**属性名称** 文件类型 **列名** 

FILTYPE

#### "文件状态"属性

#### 描述

正被监视的文件的状态

#### 类型

具有枚举值的整数。在 Tivoli Enterprise Portal 中将显示字符串。已定义的值为:OK(0)、 PERMISSION DENIED(1)、FILE DOES NOT EXIST(2)、INTERRUPTED SYSTEM CALL(4)、I/O ERROR(5)、NO SUCH DEVICE(6)、BAD FILE NUMBER(9)、OUT OF MEMORY(12)、ACCESS DENIED(13)、RESOURCE BUSY(16)、NOT A DIRECTORY(20)、IS A DIRECTORY(21)、INVALID ARGUMENT(22)、FILE TABLE OVERFLOW(23)、TOO MANY OPEN FILES(24)、TEXT FILE BUSY(26)、FILE TOO LARGE(27)、NO SPACE LEFT ON DEVICE(28)、ILLEGAL SEEK ON PIPE(29)、READ-ONLY FILE SYSTEM(30)、TOO MANY LINKS(31) 和 BROKEN PIPE(32)

#### 名称

#### 属性名称

文件状态

列名

FILSTAT

"匹配的记录数"属性

描述

此日志中已处理并与其中一种指定模式相匹配的记录的数目

类型

整数

## 名称

属性名称

匹配的记录数

列名

RECMTCH

#### "不匹配的记录数"属性

描述

发送到 UnmatchLog 的已处理记录的数目;这些记录与任何模式都不匹配

#### 类型

整数

## 名称

**属性名称** 不匹配的记录数

列名

RECUNMT

"已处理的记录数"属性

描述

自代理程序启动以来,在此日志中处理的记录数(包含并非匹配项/事件的记录)

类型

整数

名称

属性名称

已处理的记录数

列名

RECPROC

# "当前文件位置"属性

## 描述

在所监视的文件中的当前位置(以字节计)。此位置之前的数据已处理,此位置之后的数据未处理。不适用于管道。

## 类型

整数

## 名称

**属性名称** 当前文件位置

列名

OFFSET

## Current<sup>®</sup> "当前文件大小"属性

## 描述

这是受监视文件的当前大小。不适用于管道。

## 类型

整数

## 名称

属性名称

当前文件大小

## 列名

FILESIZE

## "上次修改时间"属性

## 描述

上次写入所监视文件的时间。不适用于管道。

## 类型

时间戳记

## 名称

属性名称

上次修改时间

## 列名

LASTMOD

## "代码页"属性

## 描述

所监视文件的语言代码页

## 类型

字符串

# 名称

**属性名称** 代码页

# 列名

CODEPG

# "日志文件正则表达式统计信息"属性组

"日志文件正则表达式统计信息"属性组包含的信息显示了日志文件正则表达式搜索表达式的统计信息。

您可以使用正则表达式对记录进行过滤或定义记录。此属性组显示这两种类型的相关信息。如果"结果类型"属性包含 INCLUDE 或 EXCLUDE,那么将使用过滤器对记录进行过滤。如果"结果类型"属性包含 BEGIN 或 END,那么将使用过滤器来定义记录。CPU 测量是基于操作系统所公开数据的粒度的近似值。如果正则表达式的求值时间非常短,这些测量可能会产生值 0.00。使用 CPU 时间来确定正则表达式的相对成本并优化特定正则表达式的行为。

如果您具有日志属性组并且代理程序处于 Tivoli Monitoring V6.2.1 或更高版本,那么包含"日志文件正则表达式统计信息"属性组。"**代理程序信息**"页面上将选择最低的 Tivoli Monitoring 版本。有关更多信息,请参阅<u>第 13 页的『对代理程序命名和配置』</u>。"日志文件正则表达式统计信息"属性组包含定义为 64 位数字的属性,以便能够处理较长的持续时间。对 64 位数字属性的支持由 Tivoli Monitoring V6.2.1 或更高版本提供。

以下列表包含关于"日志文件正则表达式统计信息"属性组中的每个属性的信息:

#### "节点"属性-此属性是键属性

```
描述
```

代理程序的受管系统名称。

#### 类型

字符串

#### 名称

属性名称

节点

列名

ORIGINNODE

#### "时间戳记"属性

#### 描述

收集数据时代理程序的本地时间。

#### 类型

时间

#### 名称

属性名称

时间戳记

列名

TIMESTAMP

"表名"属性 - 此属性是键属性

```
描述
```

这是日志文件属性组的名称。

```
类型
```

字符串

## 名称

属性名称

# 表名

列名

TBLNAME

## "属性名称"属性 - 此属性是键属性

## 描述

这是要对其应用此过滤器的属性的名称。

类型

字符串

#### 名称

属性名称

属性名称

列名

ATRNAME

## 过滤器编号

#### 描述

这是用于此属性的过滤器的序号(起始于零)。

#### 类型

整型(数字属性)

名称

#### 属性名称

过滤器名称

#### 列名

FLTRNUM

#### "结果类型"属性

#### 描述

结果类型可以是 INCLUDE 或 EXCLUDE, 分别用于在过滤器匹配时接受或拒绝此属性。结果类型可以是 BEGIN 或 END, 这两个值分别用于指定多行记录的记录开头和末尾。

#### 类型

具有枚举值的整数。在 Tivoli Enterprise Portal 中将显示字符串。如果使用过滤器来过滤记录,已定义的值为 INCLUDE(1) 或 EXCLUDE(2)。如果使用过滤器来定义记录,已定义的值为 BEGIN(3) 或 END(4)。

#### 名称

**属性名称** 结果类型

#### :11不子

列名 RSTTYPE

#### "平均处理器时间"属性

#### 描述

这是用于处理此属性的过滤器的平均处理器秒数。平均处理器时间是处理器总秒数除以过滤器计数而得。

#### 类型

整数(标尺)

#### 名称

属性名称

平均处理器时间

#### 列名

CPUTAVG

#### "处理器时间"属性

#### 描述

这是用于处理此属性的过滤器的总处理器秒数。处理器时间是累计时间,并且将进行截断而不是四 含五入。与 Linux /proc/<pid>/task/thread/stat 文件类似。

#### 类型

整型(计数器)

名称

属性名称

处理器时间

列名 CPUTIME

"最长处理器时间"属性

#### 描述

这是用于单一过滤器处理的最大处理器秒数。最大时间有可能为零,发生这种情况的原因是,此过 滤器从未被使用,或者每个过滤器处理耗用的时间都不足 0.01 秒。

#### 类型

整数(标尺)

#### 名称

属性名称

最长处理器时间

#### 列名

CPUTMAX

"最短处理器时间"属性

#### 描述

这是用于单一过滤器处理的最小处理器秒数。最小时间有可能为零,发生这种情况的原因是,过滤器处理的耗用时间不足 0.01 秒。

#### 类型

整数(标尺)

#### 名称

属性名称

最短处理器时间

#### 列名

CPUTMIN

## "过滤器计数"属性

## 描述

过滤器运行的次数。将此值与总处理器时间配合使用可以计算平均处理器时间。

#### 类型

整型(计数器)

## 名称

属性名称

过滤器计数

列名 COUNT

# "过滤器匹配次数"属性

描述

这是此过滤器运行且属性匹配的次数。

## 类型

整型(计数器)

## 名称

**属性名称** 过滤器匹配次数

列名 COUNTMA

#### "过滤器不匹配次数"属性

#### 描述

这是此过滤器运行且属性不匹配的次数。

## **类型** 整型(计数器)

#### 名称

#### 属性名称

过滤器不匹配次数

## 列名

COUNTUN

## "正则表达式模式"属性 - 此属性是键属性

#### 描述

这是用于匹配的正则表达式。

#### 类型

字符串

## 名称

#### 属性名称

正则表达式模式

## 列名

REGXPAT

## "上次匹配时间"属性

## 描述

上次使用此过滤器且结果匹配的时间。

## 类型

时间

## 名称

属性名称

上次匹配时间

## 列名

LASTMAT

## "上次不匹配时间"属性

## 描述

上次使用此过滤器且结果不匹配的时间。

## 类型

时间

## 名称

**属性名称** 上次不匹配时间

# 列名

LASTUMA

# 附录 D 为现有代理程序创建应用程序支持扩展

对于 IBM Tivoli Monitoring 环境,您可以构建用于分发所创建的定制工作空间、情境、查询和"执行操作" 命令的可安装包,作为对现有代理程序的应用程序支持扩展。

#### 开始之前

有关如何创建定制情境、工作空间、"执行操作"命令和查询的更多信息,请参阅<u>第181页的『第11章创</u> 建工作空间、"执行操作"命令和情境』。

#### 关于此任务

**要点:** 此任务并不是向所要构建的代理程序添加应用程序支持。要向所要构建的代理程序添加应用程序支持,请参阅第 217 页的『第 15 章 导入应用程序支持文件』。

#### 过程

- 1. 从 Agent Builder 中,选择文件 > 新建 > 其他。
- 2. 选择 Agent Builder 下的 Agent Builder 应用程序支持扩展。
- 3. 单击下一步以转到"IBM Tivoli Monitoring 应用程序支持扩展"向导的欢迎页面。
- 4. 在欢迎页面上,单击下一步。
- 5. 输入项目名称, 然后单击完成。

## 创建"应用程序支持扩展"项目

使用 Agent Builder 创建"应用程序支持扩展"项目。

#### 过程

- 1. 从 Agent Builder 中,选择文件 > 新建 > 其他。
- 2. 选择 Agent Builder 下的 Agent Builder 应用程序支持扩展。
- 3. 单击下一步以转到 IBM Tivoli Monitoring 应用程序支持扩展向导的欢迎页面。
- 4. 在欢迎页面上,单击下一步。
- 5. 输入项目名称, 然后单击完成。

## 将支持文件添加至项目

将支持文件添加至"应用程序支持扩展"项目

#### 开始之前

创建"应用程序支持扩展"项目。有关更多信息,请参阅第289页的『创建"应用程序支持扩展"项目』。

#### 过程

- 1. 右键单击"应用程序支持扩展"项目,并选择 IBM Tivoli > 导入应用程序支持扩展
- 2. 在"**导入信息**"窗口中,选择 Tivoli Enterprise Portal Server 所在主机的名称,或者单击**添加**以添加主机。
- 3. 在应用程序字段中, 输入代理程序产品代码。
- 4. 输入正在为其创建定制应用程序支持的代理程序的亲缘关系。

代理程序亲缘关系是 Tivoli Monitoring 内部标识,用于使工作空间、查询和其他的项与代理程序相关联。 它在 Tivoli Monitoring 安装中必须是唯一的。请单击**浏览**以打开"**节点类型**"窗口,并从列表中选择此信息,而不要输入此信息。

- 5. 您认为导入信息无误时,请单击完成。
- 6. 在"情境"窗口中,从"可用的情境"列表选择要导入的情境。
  单击 << 以将其添加到"选择的情境"列表,并单击确定。此时在项目下将创建一个新文件夹,其中包含安装工作空间、情境和查询所必需的文件。</p>
- 7. 在"查询"窗口中,从"可用的查询"列表选择要导入的查询。 单击 << 以将其添加到"选择的查询"列表,并单击确定。
- 8. 在"执行操作"窗口中,从"可用的执行操作"列表选择要导入的"执行操作"命令。 单击 << 以将其添加到"选择的执行操作"列表,并单击确定。代理程序的支持文件将放置在项目的相应 文件夹中。

#### 下一步做什么

您可以根据需要对任意数目的不同代理程序重复此过程。Agent Builder 会根据"应用程序支持扩展"项目中的所有支持文件创建单个安装映像。

# 生成 Application Support Extension 安装映像

生成 Application Support Extension 安装映像。

#### 过程

- 1. 右键单击"应用程序支持扩展"项目,并选择 IBM Tivoli > 创建应用程序支持扩展安装映像。
- 2. 在"应用程序支持扩展信息"窗口中,输入映像所在的目录。
- 3. Application Support Extension 必须具有自己的产品代码。请输入新代理程序的注册产品代码。您可以使用保留供 Agent Builder 使用的其中一个产品代码。允许的值是 K00-K99、K{0-2}{A-Z}和 K{4-9} {A-Z}。

**注:**这些值仅供内部使用,而不用于要共享或销售的代理程序。如果要创建与他人共享的代理程序,必须向 toolkit@us.ibm.com 发送说明,以保留产品代码。请求产品代码必须包含要构建的代理程序的描述。 然后,将为您分配、注册和返回产品代码。当您收到三个字母的产品代码时,系统将告知您如何使 Agent Builder 能够使用分配的产品代码。

- 4. 输入 Application Support Extension 的名称。
- 5. 输入 Application Support Extension 的描述。
- 6. 以 VVRRMMFF 格式输入 Application Support Extension 的版本,其中 vv = 版本号; rr = 发行版号; mm = 修改号(修订包号); ff = 临时修订号。
- 7. 单击**完成**。

# 安装 Application Support Extension

安装 Application Support Extension

#### 过程

- 1. 将映像传输到 Tivoli Enterprise Monitoring Server 和 Tivoli Enterprise Portal Server 服务器。
- 2. 要安装 Tivoli Enterprise Monitoring Server 支持,运行以下命令之一:
  - 在 Windows 上: installKXXTEMSSupport.bat
  - 在 UNIX 上: install*KXX*TEMSSupport.sh

此命令的格式如下所示:

installKXXTEMSSupport[.bat | .sh] <ITM Install Directory> [-s tems\_host]
 [-u tems\_user] \[-p tems\_password]

3. 要安装 Tivoli Enterprise Portal Server 支持,运行以下命令之一:

- 在 Windows 上: installKXXTEPSSupport.bat
- 在 UNIX 上: installKXXTEPSSupport.sh

此命令的格式如下所示:

installKXXTEPSSupport[.bat | .sh] <ITM Install Directory> [-r]

其中-r指示在安装后必须重新启动 Tivoli Enterprise Portal Server

# 将解决方案安装项目转换为"应用程序支持扩展"项目

将现有解决方案安装项目转换为"应用程序支持扩展"项目

#### 关于此任务

如果存在要转换为"应用程序支持扩展"项目的现有**解决方案安装项目**,请完成下列步骤: **注:**在**解决方案安装项目**中,将仅迁移支持文件。

#### 过程

1. 右键单击解决方案安装项目并选择 IBM Tivoli > 转换解决方案安装项目。

- 2. 输入新的"应用程序支持扩展"项目的名称,或者从列表中选择现有的项目
- 3. 单击**完成**。

# 附录 E 生成 Cognos 数据模型

Agent Builder 可以为每个代理程序生成一个 Cognos 数据模型。使用该数据模型可将代理程序信息导入到 Cognos Framework Manager 中,以便创建报告。

您可以在 Framework Manager 中打开并查看此 Cognos 数据模型,而 Framework Manager 将构建要发布到 Tivoli Common Reporting 的模型包。在发布之前,还可以在 Framework Manager 中对此数据模型进行定制 或修改。

创建报告后,Agent Builder 还允许将最终的报告包导入到 Agent Builder 项目中。此功能使您将来可以使用 代理程序包中已包含的报告来生成代理程序项目。作为代理程序安装映像组成部分打包的报告可以导入到生 产环境中的 Tivoli Common Reporting 中。

注: 在本文档中, 请注意下列约定:

- ·Kxx 或 kxx 是指对代理程序指定的产品代码,例如 k99。
- · dbType 是指 Tivoli Data Warehouse 所使用的数据库,例如 Db2。

# 生成 Cognos 数据模型的先决条件

在生成 Cognos 数据模型之前,请完成下列任务

#### 关于此任务

注:

·这些步骤仅需完成一次,这是因为,将来使用 Agent Builder 生成的所有数据模型都将使用此环境。

·您最好创建一个隔离的开发环境,用于测试代理程序和创建报告。

#### 过程

- 1. 安装并配置 第 293 页的『Tivoli Data Warehouse』。
- 2. 在 Tivoli Data Warehouse 中创建表及过程。
  - a) <u>第 293 页的『在 Tivoli Data Warehouse 中创建表和过程』</u>.
  - b) 第 296 页的『以 Tivoli Reporting and Analytics Model 填充 Tivoli Data Warehouse』.
- 3. 安装并配置 第 296 页的『Tivoli Common Reporting』。
- 4. 安装并配置 第 297 页的『Framework Manager』。

### Tivoli Data Warehouse

关于 Tivoli Data Warehouse。

要创建报告,您需要在环境中安装并配置 Tivoli Data Warehouse、Warehouse Proxy Agent 和 Summarization and Pruning Agent。有关更多信息,请参阅《IBM Tivoli Monitoring 安装与设置指南》。

#### 在 Tivoli Data Warehouse 中创建表和过程

在 Tivoli Data Warehouse 中创建或变更 ManagedSystem 表和存储过程

#### 关于此任务

生成的 Cognos 数据模型包含用于定义 ManagedSystem 维度的 ManagedSystem 表。ManagedSystem 维度支持创建可以关联受管系统的报告。例如,对于子节点代理程序,可以使用该维度来确定特定代理程序实例的子节点。

ManagedSystem 表并非由 Tivoli Data Warehouse 创建。因此,在 Agent Builder 中生成代理程序时,将为 每个数据库平台生成 SQL 脚本,这些脚本将执行下列操作:

- · 创建 ManagedSystem 表。如果 Tivoli Data Warehouse 中不存在该表,请使用此脚本。
- ·编辑 ManagedSystem 表。如果 Tivoli Data Warehouse 中存在该表,请使用此脚本。其他报告产品可以 创建 ManagedSystem 表,但它们创建的表中没有全部的必需列。
- ·创建一个根据 Tivoli Data Warehouse 中的表填充 ManagedSystem 表的存储过程。

这些脚本只需运行一次。

#### 运行 Db2 脚本以在 Tivoli Data Warehouse 中创建表和过程

对于 Db2 数据库,请使用以下脚本在 Tivoli Data Warehouse 中创建表

#### 开始之前

适用于 Db2 的脚本位于以下目录中:

reports/db2/Kxx/reports/cognos\_reports/itmkxx/db\_scripts

#### 过程

- 生成的脚本(create\_table.sql、alter\_table.sql和 create\_procedure.sql)全都使用 *itmuser* 作为 Tivoli Data Warehouse 用户标识。在您的环境中,如果 Tivoli Data Warehouse 用户标识 并非 *itmuser*,请将出现的所有 *itmuser* 更改为正确的用户标识。
- 2. 以 Tivoli Data Warehouse 用户身份连接到 Tivoli Data Warehouse:

db2 connect to <Tivoli Data Warehouse alias name> user <Tivoli Data Warehouse user id> using <password>

3. 确定 ManagedSystem 表是否存在:

db2 "select count(\*) from sysibm.systables where name = 'MANAGEDSYSTEM'
and creator=upper ('<Tivoli Data Warehouse user id>')"

- 4. 创建或变更该表。
  - ·如果以上查询返回1,那么表明该表已存在。请运行变更脚本:

db2 -tvf alter\_table.sql

· 如果以上查询返回 0, 那么表明该表不存在。请运行创建脚本:

db2 -tvf create\_table.sql

5. 运行脚本以创建存储过程:

db2 -td@ -f create\_procedure.sql

#### 运行 Oracle 脚本以在 Tivoli Data Warehouse 中创建表和过程

对于 Oracle 数据库,请使用以下脚本在 Tivoli Data Warehouse 中创建表

#### 开始之前

适用于 Oracle 的脚本位于以下目录中:

reports/oracle/Kxx/reports/cognos\_reports/itmkxx/db\_scripts

#### 过程

- 1. 生成的脚本(create\_table.sql、alter\_table.sql和 create\_procedure.sql) 全都使用 *itmuser* 作为 Tivoli Data Warehouse 用户标识。在您的环境中,如果 Tivoli Data Warehouse 用户标识 并非 *itmuser*,请将出现的所有 *itmuser* 更改为正确的用户标识。
- 2. 启动 sqlplus:

sqlplus <IBM Tivoli Monitoring user ID>/<password>@ <Tivoli Data Warehouse SID>

3. 确定 ManagedSystem 表是否存在:

```
select count(*) from user_tables where table_name = 'MANAGEDSYSTEM';
```

- 4. 创建或变更该表。
  - ·如果以上查询返回1,那么表明该表已存在。请运行变更脚本:

@<path to alter\_table.sql>;

·如果以上查询返回 0,那么表明该表不存在。请运行创建脚本:

@<path to create\_table.sql>;

5. 运行脚本以创建存储过程:

@<path to create\_procedure.sql>;

#### 运行 SQL Server 2005 和 2008 脚本以在 Tivoli Data Warehouse 中创建表和过程

#### 开始之前

适用于 SQL Server 的脚本位于以下目录中:

reports/mssql/Kxx/reports/cognos\_reports/itmkxx/db\_scripts

#### 过程

- 生成的脚本(create\_table.sql、alter\_table.sql和 create\_procedure.sql)全都使用 *itmuser* 作为 Tivoli Data Warehouse 用户标识。在您的环境中,如果 Tivoli Data Warehouse 用户标识 并非 *itmuser*,请将出现的所有 *itmuser* 更改为正确的用户标识。
- 2. 确定 ManagedSystem 表是否存在:

osql -S <Server> -U <Tivoli Data Warehouse user ID> -P <password> -d <Tivoli Data Warehouse database name> -Q "Select count(\*) from INFORMATION\_SCHEMA.TABLES where table\_name = 'ManagedSystem'"

#### 3. 创建或变更该表。

·如果以上查询返回1,那么表明该表已存在。请运行变更脚本:

osql -S <Server> -U <Tivoli Data Warehouse user ID> -P <password> -d <Tivoli Data Warehouse database name> -I -n -i <path to alter\_table.sql>

·如果以上查询返回 0,那么表明该表不存在。请运行创建脚本:

osql -S <Server> -U <Tivoli Data Warehouse user ID> -P <password> -d <Tivoli Data Warehouse database name> -I -n -i <path to create\_table.sql>

#### 4. 运行脚本以创建存储过程:

osql -S <Server> -U <Tivoli Data Warehouse user ID> -P <password> -d <Tivoli Data Warehouse database name> -I -n -i <path to create\_procedure.sql>

#### 以 Tivoli Reporting and Analytics Model 填充 Tivoli Data Warehouse

使用提供的数据库脚本来填充 Tivoli Data Warehouse

#### 关于此任务

Tivoli Reporting and Analytics Model (TRAM) 包含所有报告包通用的基本知识集。TRAM 通过每个数据库所 特有的一组脚本进行安装。用于填充每个受支持数据库的必需脚本在代理程序安装映像的 reports 目录中提 供。使用以下过程可以在 Tivoli Data Warehouse 中创建 Tivoli Reporting and Analytics Model 公共维度。

## 过程

- 1. 浏览到 Tivoli Reporting and Analytics Model 数据库脚本。
- 2. 解压缩代理程序包。
  - · 在 Windows 系统上,代理程序包是 kxx.zip。
  - · 在 Linux 和 UNIX 系统上,代理程序包是 kxx.tgz。
- 3. 转到相应的数据库脚本。
  - · Db2 脚本位于代理程序包中的以下位置:

reports/db2/Kxx/reports/cognos\_reports/itmkxx/db\_scripts

· Oracle 脚本位于代理程序包中的以下位置:

reports/oracle/Kxx/reports/cognos\_reports/itmkxx/db\_scripts

· Microsoft SQL Server 脚本位于代理程序包中的以下位置:

reports/mssql/Kxx/reports/cognos\_reports/itmkxx/db\_scripts

- 4. 运行数据库脚本,以在 Tivoli Data Warehouse 中生成公共维度。每个脚本集都提供有包含用法指示信息的自述文件。
- 5. 确认这些脚本已将下列各表添加到 Tivoli Data Warehouse:

"Computer System", WEEKDAY\_LOOKUP, MONTH\_LOOKUP, TIMEZONE\_DIMENSION, TIME\_DIMENSION

## **Tivoli Common Reporting**

Tivoli Common Reporting 包含 Cognos Business Intelligence 引擎,其中包含用于帮助您创建代理程序报告的元素。

您必须安装 Tivoli Common Reporting 并为其配置一个连接到 Tivoli Data Warehouse 的数据源。

#### 安装 Tivoli Common Reporting

您必须安装 Tivoli Common Reporting。支持 V1.3、V2.1、V2.1.1 或更高版本。有关安装 Tivoli Common Reporting 的信息,请参阅<u>安装 Tivoli Common Reporting</u>。

#### 配置 Tivoli Common Reporting

您必须配置 Tivoli Common Reporting。有关配置 Tivoli Common Reporting 的信息,请参阅配置 IBM Tivoli Common Reporting。

在 Tivoli Data Warehouse 与 Tivoli Common Reporting 之间创建数据源。有关更多信息,请参阅<u>配置数据库</u> <u>连接</u>。单击相应的数据库类型。请记录对该数据源指定的名称。缺省值为 **TDW**。

**注:**数据源名称必须与"Cognos 信息"页面的数据源字段中的名称匹配。有关"Cognos 信息"页面的更多 信息,请参阅 <u>第 27 页的『Cognos 信息』</u>。

## **Framework Manager**

Framework Manager 是随 Tivoli Common Reporting 应用程序一起提供的应用程序,但必须单独安装和配置。

Framework Manager 用于查看和修改数据模型以及将数据模型发布到 Tivoli Common Reporting

#### 安装 Framework Manager

您必须安装 Framework Manager。支持 V8.4、V8.4.1 或更高版本。

Framework Manager 随 Tivoli Common Reporting 一起提供,但必须手动安装。Tivoli Common Reporting 1.3 随附了 Framework Manager 8.4。Tivoli Common Reporting 2.1 和 2.1.1 随附了 Framework Manager 8.4.1。有关安装 Framework Manager 的信息,请参阅 *Tivoli Common Reporting User's Guide* 中的 Installing Framework Manager。

#### 配置 Framework Manager

您必须配置 Framework Manager。有关配置 Framework Manager 的信息,请参阅 Tivoli Common Reporting User's Guide 中的 Configuring Framework Manager。

# 创建报告

使用 Framework Manager 发布代理程序模型,并使用 Report Studio 开始创建报告。

#### 开始之前

代理程序完成后,它必须安装到 Tivoli Monitoring 环境中。另外,必须配置代理程序历史记录收集,并使该 代理程序至少运行一个仓库上载时间间隔。必须配置摘要,并且 Tivoli Monitoring 中选择的摘要设置选项必 须与 Agent Builder 中选择的摘要选项完全相同。将代理程序的数据上载到仓库之后,必须至少运行一次 Summarization and Pruning Agent。

1. 安装、配置并启动代理程序。

2. 对于每个要为其创建报告的属性组,请创建相应的历史记录收集并将其分发到代理程序。

注:仓库上载时间间隔缺省为"每天"。但是,您可能希望缩短此时间间隔。

有关配置历史记录收集的信息,请参阅 IBM Tivoli Monitoring Administrator's Guide 中的 Managing historical data。

3. 在 Tivoli Monitoring 中,针对所有在步骤 2 创建了历史记录收集的属性组,配置摘要。

注: 当您配置历史记录收集及摘要时,必须等待足够长的时间才能得到摘要表中的数据。

注: 缺省情况下, Summarization and Pruning Agent 配置为每天凌晨 2 点运行一次。您可能想更改此设置。例如,可以将其配置为每小时运行一次。有关配置 Tivoli Data Warehouse 的信息,请参阅 *IBM Tivoli Monitoring Installation and Setup Guide* 中的 <u>Setting up data warehousing</u>。

## 关于此任务

在 Agent Builder 中生成代理程序将创建完整的 Framework Manager 项目,其中包含数据模型以及 Framework Manager 项目文件。Framework Manager 可以直接打开项目文件,这将打开数据模型以进行修改、定制或发布。

#### 过程

**注:** 生成的代理程序数据模型包含每个属性组的所有摘要时间维度:每小时、每日、每周、每月、每季度和每年。只有为代理程序配置摘要及修剪之后,这些维度才会存在于代理程序的 Tivoli Data Warehouse 中。如果已选择这些维度,并且 Summarization and Pruning Agent 已创建并填充了表,也会如此。您可以定义使用不存在的维度的报告,并将其发布至 Tivoli Common Reporting。此类报告直到 Summarization and Pruning Agent 创建摘要表之后才有效。

1. 在 Framework Manager 中打开代理程序数据模型:

- a) 打开 Framework Manager。
- b) 在"欢迎"页面中, 单击打开项目。

提示:如果您是在 Framework Manager 中,请单击文件菜单中的打开。

c) 浏览至代理程序数据模型。

· 对于 Db2:

reports/db2/Kxx/model/

·对于 Oracle:

reports/oracle/Kxx/model/

· 对于 Microsoft SQL Server:

reports/mssql/Kxx/model/

#### d) 选择代理程序项目文件 Kxx.cpf。

🗅 😤 🔲 Mattada 🔍 Res 🕅 🗙
Composition         Contraction           Projects         Wind Bears           Wind Bears         Wind Bears <t< th=""></t<>

#### 图 50. 选择代理程序项目文件

注: 在 Framework Manager 中打开代理程序项目时,代理程序名称将在"最近的项目"下列出。 2. 填写"受管系统表"。有关更多信息,请参阅<u>第 302 页的『填充 ManagedSystem 表』</u>

- 3. 使用 Framework Manager 将代理程序模型发布至 Tivoli Common Reporting
  - a) 打开 Framework Manager。
  - b) 打开代理程序项目。
  - c) 在导航树中, 展开包。
  - d) 右键单击该代理程序包, 然后选择发布包。



图 51. 选择发布包

4. 使用 Report Studio 创建新的报告或模板。

a) 登录 Tivoli Common Reporting。

b) 浏览至"公用文件夹",在导航面板中展开报告,并选择 Common Reporting。



图 52. 选择 Common Reporting

- c) 从提供的列表中选择 Tivoli Monitoring 代理程序。
- d) 通过单击"启动"菜单并选择 Report Studio 或 Query Studio,打开报告创建工具。

🥭 Tivoli Integrated Portal - Windows Interne	t Explorer	
😋 💿 🗢 🙋 https://localhost:16316/bm/con	iale/secure/securelogon.do 🔄 🔒 🖹 🆅 🗙 🔀	ing 🖉 🔹
tile Edit View Favorites Loois Help		
🚖 Favorites 🛛 🍰 🙋 Suggested Sites 👻 🙋 Pa	ee Hotmail 🙋 Web Size Galery 🝷	
💋 Tivoli Integrated Portal	🔂 • 6	🛛 + 🖃 🖶 + Bage + Safety + Tooks + 😥 +
Tivoli. View: All tasks 💌	Welcome tipadmin	Help   Logout IBM.
•	Common Repo ×	Select Action 💌
- Welcome - My Startup Pages	Work with reports	
Security	IBM Cognos Connection tipadmin 🧑	Q v 👘 v 🖄 v Launch v 🕢 v
Users and Groups	Public Folders     My Folders	Report Studio
Troubleshooting	Public Folders > IBM Tivoli Monitoring for Windows OS	😂 👫 Drill-through Definitions 🖡 🖓
Common Reporting		Entration IBM Country Administration
Settings	□   Name ↔   Modified ↔	Actions
	No entries.	
Done		Jocal intranet 🛛 🐔 📲 🔩 LOO% 🔹 🦯

图 53. 选择 Report Studio

# 下一步做什么

您可以使用 Report Studio 来创建新的报告或模板,也可以对现有的报告或模板进行修改。



#### 图 54. Report Studio

有关更多信息,请参阅 IBM Knowledge Center 上的 Tivoli Common Reporting 主题集合。

## 填充 ManagedSystem 表

ManagedSystem 表通过 kqz\_populate\_msn 存储过程进行填充。

有关更多信息,请参阅<u>第 303 页的『运行 Db2 存储过程』</u>。此过程必须定期运行,以使 ManagedSystem 表包含受管系统名称的最新列表。

此存储过程将读取 Tivoli Data Warehouse 中的下列历史表(如果它们存在):

- ·代理程序的 Performance Object Status 表
- ·代理程序的可用性表。用于监视进程或服务的代理程序具有可用性表。
- ·代理程序的发现表。子节点代理程序将创建发现表。

必须对特定一组属性组启动历史数据收集。将生成一组脚本,这些脚本将对这些属性组创建并启动历史数据 收集。如果您不希望使用脚本,那么属性组列表将列示在脚本的注释头块中。

创建了样本脚本,用于说明必须启用历史数据收集的表:

・reports/configuretdw.sh

・reports/configuretdw.bat

下表描述必需的自变量:

注:必须指定 -n 或 -m, 但不能同时指定这两者。

表 45. 必需自变量			
参数	描述		
-h candle_home	Tivoli Monitoring 安装路径。		
-u teps_user	您创建历史数据收集时用于进行登录的 Tivoli Enterprise Portal Server 用户。		
-n tems_name	必须在其中启动收集的 Tivoli Enterprise Monitoring Server。通过使用以空格分隔的列表,可以指定多个 Tivoli Enterprise Monitoring Server。如果指定了多 个 Tivoli Enterprise Monitoring Server,请将列表括 在引号内。例如, -n "tems1 tems2"		
-m managed_system_group_or_managed_system	必须对其启动收集的受管系统组或受管系统的名称。 可以使用以空格分隔的列表来指定多个受管系统组或 受管系统。如果指定了多个受管系统组或受管系统, 请将列表括在引号内。例如, -m "msg1 msg2"		

下表描述可选的自变量:

表 46. 可选自变量			
参数	描述		
-s teps_host	Tivoli Enterprise Portal Server 的主机名或 IP 地址。如果未指定,那么将使用缺省值 localhost。		
-p teps_password	使用 - u 选项指定的 Tivoli Enterprise Portal Server 用户的密码。如果未指定,那么脚本会提示您输入密码		
-c historical_collection_interval	启动历史数据收集时使用的历史数据收集时间间隔。 如果未指定,那么将使用缺省值1h(1小时)。有 效值为:15m、30m、1h、12h或1d,其中m是分 钟,h是小时,d是天。		
-r pruning_interval	用于历史数据的修剪时间间隔。必须对历史数据进行 修剪,以使表不会持续增大。如果未指定,那么将使 用缺省值 2d(2 天)。使用 d 表示天,m 表示月,y 表示年。		

启动历史数据收集后,必须定期运行 kqz\_populate\_msn 存储过程。此存储过程定期运行,以使 ManagedSystem 表包含 Tivoli Monitoring 环境中受管系统的最新列表。

#### 运行 Db2 存储过程

对 Db2 运行存储过程。

#### 关于此任务

请执行下列步骤,以便对 Db2 运行存储过程:

## 过程

1. 以仓库用户身份连接至 Tivoli Data Warehouse 数据库:

connect to <Tivoli Data Warehouse database alias> user <Tivoli Data Warehouse user id> using <password>

2. 运行存储过程:

```
db2 "call <Tivoli Data Warehouse schema>.kqz_populate_msn
('<three letter product code for the agent>')"
```

#### 运行 Oracle 存储过程

在 Oracle 上运行存储过程。

#### 关于此任务

请执行下列步骤,以便对 Oracle 运行存储过程:

#### 过程

1. 启动 sqlplus:

sqlplus <Tivoli Data Warehouse user id>/<password>@
<Oracle SID>

2. 运行存储过程:

execute kqz\_populate\_msn('<three letter product code for the agent>');

#### 在 SQL Server 2005 和 2008 上运行存储过程

在 SQL Server 上运行存储过程。

#### 关于此任务

请执行下列步骤, 以便对 SQL Server 2005 和 2008 运行存储过程:

#### 过程

运行存储过程:

```
osql -S <server> -U <Tivoli Data Warehouse id> -P
<Tivoli Data Warehouse password> -d
<Tivoli Data Warehouse database name> -Q "EXEC
[<Tivoli Data Warehouse schema>].[kqz_populate_msn]
@pv_productcode = N'<three letter product code>'"
```

#### 从 Tivoli Common Reporting 中导出报告和数据模型

从 Tivoli Common Reporting 中导出报告和数据模型。

#### 过程

- 1. 登录到 Tivoli Common Reporting。
- 2. 转至"公用文件夹",在导航面板中的报告下,选择 Common Reporting。
- 3. 在"使用报告"部分中,单击启动菜单并选择 IBM Cognos 管理。
- 4. 单击配置选项卡。
- 5. 单击内容管理。

🖉 Tivoli Integrated Portal - Windows Int	ernet Explorer				
C C F I https://localhost:16316/ibm/console/login.do?action=secure		ure	🔄 🔒 🕾 🐓 🗙 🔽 Bing 🖉		
<u>File Edit View Favorites Tools Help</u>	)				
🚖 Favorites 🛛 🤹 🙋 Suggested Sites 🔹 👔	🔊 Free Hotmail 🛛 🙋 Web Slic	e Gallery 🔻			
🦉 Tivoli Integrated Portal				🐴 • 🔂 🛪 🖃 🖶 • Page • Safety • Tools • 🔞	
Tivoli. View: All tasks 💌				Help Logout IBM	
+ =	Common Repo.	×		Select Action	
Welcome					
My Startup Pages	Work with repo	orts			
Security	IBM Cognos Adr	ninistration		tipadmin 🔅 🏠 🗙 🖉 🗸 Launch 🗸 🥑	
Users and Groups     Troubleshooting     Reporting     Common Reporting     Settings	Status	Security	Configuration		
	Data Source Cor	Data Source Connections     Content Administration		📫 📽 🖾 🖄 🕼 🕂 🗈 🛍 🗶 💭 Q	
	🐻 Content Admi				
	Distribution Lists	Distribution Lists and Contacts		Entries: - O M M M M	
an 11 - 1999.	Printers	Printers		Modified ⊕   Actions	
	Styles			No entries.	
	Portlets				
	2. Dispatchers and	Services			
			Last refresh time: June 3, :	2011 7:55:51 AM	
	17.1			💦 📢 Local intranet 🛛 🖓 🔹 🔍 100% 🔹	

图 55. "内容管理"选项卡

6. 单击新建导出图标以导出新包。

7. 对这个包进行命名。(可选)您可以添加屏幕提示和描述。

8. 选择选择公用文件夹和目录内容。

9. 在"公用文件夹"对话框中,单击添加链接。

10. 将代理程序包移到选定的条目。

11. 在向导的最后一页上,选择仅保存。在向导完成时,此报告包会列示在"内容管理"选项卡上。

12. 在"内容管理"选项卡上,单击绿色箭头(运行)以创建压缩.zip文件。

🌈 Tivoli Integrated Portal - Windows Internet Explorer 📃 🛛 🗙			
COC - Ittps://localhost:16316/ibm/con	isole/secure/securelogon.do		№ * × 5 Bing
Eile Edit View Favorites Tools Help			
🚖 Favorites 🛛 🚔 🙋 Suggested Sites 👻 🔊 Fr	ree Hotmail 💈 Web Slice Gallery 🔻		
🏉 Tivoli Integrated Portal			🏠 🔹 🔝 👻 🖃 🖶 👻 Bage 🔹 Safety 🔹 Tools 🔹 🔞 🔹
Tivoli. View: All tasks 💌	V	Velcome tipadmin	Help Logout IBM.
• •	Common Repo ×		Select Action 💌
<ul> <li>Welcome</li> <li>My Startup Pages</li> </ul>	Work with reports		
Security	IBM Cognos Administration		tipadmin 🔗 🏠 - 🖓 - 🖓 - Launch - 🥝 -
<ul> <li>Users and Groups</li> <li>Tooublockersting</li> </ul>	Status Security	Configuration	
Reporting	Data Source Connections	Administration	😂 📽 🛃 🦝 🕼 v 🗈 💼 🔺 💭 🔍
- Common Reporting	🔓 Content Administration		
Settings	Distribution Lists and Contacts		Entries: 1 - 1 🚺 🕅
J. J	Printers	□   Name ≑	Modified  Actions
	S Styles	Last refresh time: August 9,	2011 1:40:11 PM
	Portlets	L	
	Control Contro		
,			💦 📢 Local intranet 🌾 🔹 100% 👻 🏸

图 56. 列出代理程序包的 "内容管理" 选项卡

#### 结果

导出过程创建的压缩.zip文件位于部署目录中。

·Tivoli Common Reporting V1.3 的目录路径如下所示:

C:\IBM\tivoli\tip\products\tcr\Cognos\c8\deployment

·Tivoli Common Reporting V2.1 或更高版本的目录路径如下所示:

C:\IBM\tivoli\tipv2Components\TCRComponent\cognos\deployment

#### 下一步做什么

有关导出报告的更多信息,请参阅 *Tivoli Common Reporting User's Guide* 中的 <u>Exporting Cognos report</u> <u>packages</u>。

## 将报告导入到 Agent Builder 中

从 Tivoli Common Reporting 中导出报告包之后,可以将其导入到 Agent Builder 项目中。然后,可以在代理 程序安装映像中包括报告包。

#### 过程

- 1. 在 Agent Builder 中,右键单击代理程序项目。
- 2. 选择 IBM > 导入报告包。
- 3. 在"导入报告包"窗口中,选择创建报告包时所要基于的数据库类型。
- 4. 输入报告包的标准路径,或者单击浏览以选择报告包。
- 5. 单击确定。

6. 报告包现在显示在代理程序项目中的 reports/dbtype 目录下。

注:如果创建特定于数据库的报告包,必须将每个包导入到 Agent Builder 中。

## 将代理程序包中的报告安装到 Tivoli Common Reporting 中

将代理程序中的报告包导入到 Tivoli Common Reporting 中

## 过程

1. 按照向导中的步骤,从代理程序映像导入新的包。

在代理程序映像中, 报告位于: reports/dbType/Kxx/reports/cognos\_reports/itmkxx/packages

- 2. 将报告压缩 zip 文件复制到 Tivoli Common Reporting 部署目录。
  - Tivoli Common Reporting V1.3 的目录路径为: C:\IBM\tivoli\tip\products\tcr\Cognos \c8\deployment
  - Tivoli Common Reporting V2.1 或更高版本的目录路径为: C:\IBM\tivoli\tipv2Components \TCRComponent\cognos\deployment
- 3. 登录到 Tivoli Common Reporting。
- 4. 转至"公用文件夹",在导航面板中的报告下,选择 Common Reporting。
- 5. 在"使用报告"部分中,单击启动菜单并选择 IBM Cognos 管理。
- 6. 转至配置选项卡并打开内容管理部分。
- 7. 单击新建导入以创建包导入。
- 8. 选择代理程序的报告包。
- 9. 选择要导入的公共文件夹。
- 10. 选择 Save。
- 11. 单击绿色(运行)箭头以导入。

## 结果

有关更多信息,请参阅《Tivoli Common Reporting 用户指南》中的『登录报告界面』。
# 附录 F ICU 正则表达式

ICU 正则表达式实施细节的描述。

此引用内容摘自 ICU User Guide。此内容描述 ICU 正则表达式实施细节。如果您在使用 Agent Builder 正则 表达式功能,那么此信息很重要,因为不同编程语言实施正则表达式的方式略有不同。

表 47. 正则表达式元字符		
字符	描述	
\a	与响铃符 \u0007 匹配	
\A	在输入内容的开头处匹配。与 ^ 的不同之处是 \A 不 在输入内容中的换行后匹配。	
\b, 在[Set]外	如果当前位置是单词边界就匹配。边界在单词 (\w) 和非单词 (\W) 字符间的过渡处出现,忽略组合标 记。有关单词边界的更多信息,请参阅"ICU 边界分 析"。	
\b, 在[Set]内	与退格符 \u0008 匹配。	
\В	如果当前位置不是单词边界就匹配。	
\cX	与Ctrl-X字符匹配。	
\d	与具有 Unicode 常规类别 Nd(N 表示数字, d 表示 十进制数字)的任何字符匹配。	
\ D	与任何非十进制数字的字符匹配。	
\e	与转义符 \u001B 匹配。	
\E	终止由 \Q \E 引起来的序列。	
\f	与换页符 \u000C 匹配。	
\G	如果当前位置是先前匹配的末尾就匹配。	
\n	与换行符 \u000A 匹配。	
\N{UNICODE CHARACTER NAME}	与指定字符匹配。	
\p{UNICODE PROPERTY NAME}	与任何具有指定 Unicode 属性的字符匹配。	
\P{UNICODE PROPERTY NAME}	与不具有指定 Unicode 属性的任何字符匹配。	
١Q	在 \E 之前的所有字符两边加引号。	
\r	与回车符 \u000D 匹配。	
\s	与空格字符匹配。空格定义为 [\t\n\f\r \p{Z}]。	
\S	与非空格字符匹配。	
\t	与横向制表符 \u0009 匹配。	
\uhhhh	与具有十六进制值 hhhh 的字符匹配。	
\Uhhhhhhh	与具有十六进制值 hhhhhhhh 的字符匹配。必须正好提供八个十六进制数字,即使最大的 Unicode 代码点为 \U0010ffff。	

表 47. 正则表达式元字符 (续)	
字符	描述
\w	与单词字符匹配。单词字符为 [\p{L1}\p{Lu} \p{Lt}\p{Lo}\p{Nd}]。
\W	与非单词字符匹配。
\x{hhhh}	与具有十六进制值 hhhh 的字符匹配。可以提供1到 6 个十六进制数字。
\xhh	与具有2位数十六进制值 hh 的字符匹配。
\X	与语素簇匹配。
	如果当前位置在输入的结尾处、而在最后一行的终止符(如果存在)之前就匹配。
\z	如果当前位置在输入内容结尾处就匹配。
\n	反向引用。与和第 n 个捕获组匹配的任何对象匹 配。n 必须为 > 1 且 < 模式中捕获组总数的数字。
	注:ICU 正则表达式中不支持八进制转义符(如 \012)。
[pattern]	与集内的任意一个字符匹配。有关模式中可能出现的 内容的完整描述,请参阅 UnicodeSet
•	与任意字符匹配。
^	在行开头处匹配。
\$	在行结尾处匹配。
	在以下字符两边加引号。必须加引号才能被视为字面 值的字符有: * ? + [() {} ^ \$   \. /

表 48. 正则表达式运算符	
运算符	描述
	交替。A   B 与 A 或 B 匹配。
*	匹配0次或0次以上。匹配次数尽可能地多。
+	匹配1次或1次以上。匹配次数尽可能地多。
?	匹配0次或1次。首选1次。
٤u}	精确匹配 n 次
{n,}	至少匹配n次。匹配次数尽可能地多。
{n,m}	匹配 n 到 m 次。匹配次数尽可能地多,但不超过 m 次。
*?	匹配0次或0次以上。匹配次数尽可能地少。
+?	匹配1次或1次以上。匹配次数尽可能地少。
??	匹配0次或1次。首选0次。
{n}?	精确匹配 n 次

表 48. 正则表达式运算符 (续)		
运算符	描述	
{n,}?	至少匹配 n 次,但是不超过整体模式匹配所需的次数	
{n,m}?	匹配 n 到 m 次。匹配次数尽可能地少,但不少于 n 次。	
*+	匹配 0 次或 0 次以上。首次遇到时,匹配的次数尽可能地多,即使总体匹配失败,重试次数也不减少 (占有匹配)	
++	匹配1次或1次以上。占有匹配。	
?+	匹配 0 次或 1 次。占有匹配。	
{n}+	精确匹配 n 次	
{n,}+	至少匹配 n 次。占有匹配。	
{n,m}+	匹配 n 到 m 次。占有匹配。	
( )	捕获括号。与括起的子表达式匹配的输入范围在该匹配后可用。	
(?: )	非捕获括号。将包含的模式进行分组,但是不提供对 匹配文本的捕获。比捕获括号的效率更高。	
(?> )	原子匹配括号。与括起子表达式的首次匹配是唯一尝 试的一次匹配。如果此次匹配未导致整体模式匹配, 备份与"(?>"前面的位置匹配的搜索。	
(?# )	自由格式注释(?# comment )。	
(?= )	先行断言。如果括起的模式在当前输入位置匹配而没 有提前输入位置,那么为 True。	
(?! )	否定先行断言。如果括起的模式在当前输入位置不匹配,那么为 True。不提前输入位置。	
(?<= )	后顾断言。如果括起的模式与当前输入位置前面的文本匹配,那么为 True。匹配的最后一个字符正好是位于当前位置前面的输入字符。不变更输入位置。后顾模式匹配的可能字符串长度必须受限(无*或+运算符)。	
(? )</th <th>否定后顾断言。如果括起的模式与当前输入位置前面 的文本不匹配,那么为 True。匹配的最后一个字符 正好是位于当前位置前面的输入字符。不变更输入位 置。后顾模式匹配的可能字符串长度必须受限(无 * 或 + 运算符)。</th>	否定后顾断言。如果括起的模式与当前输入位置前面 的文本不匹配,那么为 True。匹配的最后一个字符 正好是位于当前位置前面的输入字符。不变更输入位 置。后顾模式匹配的可能字符串长度必须受限(无 * 或 + 运算符)。	
(?ismx-ismx: )	标志设置。在启用或禁用指定标志的情况下对括起的表达式求值。	
(?ismx-ismx)	标志设置。更改标志设置。更改应用于设置后面的模式部分。例如,(?i)更改为不区分大小写的匹配。	

# 替换文本

"查找与替换"操作的替换文本可以包含对来自查找的捕获组文本的引用。引用的形式为 \$n, 其中 n 是捕获组数量。

表 49. 替换文本字符	
字符	描述
\$n	用位置捕获组 n 的文本替换 \$n。n 必须 >= 0,并且 不大于捕获组数量。\$ 后不跟数字不表示任何特殊意 义,在替换文本中将按原样显示为 \$。
λ	将此字符视为字面值,不表示任何特殊含义。替换文 本中的反斜杠转义仅针对"\$"和"\"是必需的, 但可以用于任何其他字符,而不会产生反作用。
\$@n	用捕获组 n 的文本替换与捕获组 n 匹配的正则表达 式。n 必须 >= 0,但不大于捕获组数量。\$0 后不跟 数字不表示任何特殊意义,在替换文本中将按原样显 示为 \$0。
\$#n	用匹配捕获组 n 的文本替换 \$#n。n 必须 >= 0,并 且不得大于匹配的捕获组数量。\$# 后不跟数字不表 示任何特殊意义,在替换文本中将按原样显示为 \$#。

# 标志选项

以下标志用于控制正则表达式匹配的各个方面。可以在将表达式编译为 RegexPattern 对象时指定标志值。 或者,可以使用 (?ismx-ismx) 模式选项在模式自身内部指定。

表 50. 标志选项		
标志(模式)	标志(API 常量)	描述
i	UREGEX_CASE_INSENSITIVE	如果设置了此标志,将以不区分大 小写的方式进行匹配。
x	UREGEX_COMMENTS	如果设置了此标志,那么可以在模式中使用空格和 #comments。
S	UREGEX_DOTALL	如果设置了此标志,那么模式中的 "."将与输入文本中的行终止符 匹配。缺省情况下不匹配。文本中 的回车符/换行对等效于单个行终 止符,并且与 RE 模式中的单个 "."匹配
m	UREGEX_MULTILINE	控制模式中"^"和"\$"的行 为。缺省情况下,这些模式分别仅 在输入文本的开头和结尾处匹配。 如果设置了此标志,那么"^"和 "\$"还在输入文本内部每一行的 开头和结尾处匹配。

# 附录 G 创建非代理程序文件捆绑包

您能够创建可以放入 Tivoli Monitoring 库的文件捆绑包。然后,在您的环境中,可以将这些文件捆绑包部署 到目标系统。

## 关于此任务

借助此功能,您可以采用远程方式来配置没有远程配置选项的产品。要使用此功能,请将进行了预填充的配置文件放入库中,并将其发送到所需的系统。

#### 过程

- 1. 从 Agent Builder 中,选择文件 > 新建 > 其他。
- 2. 在 Agent Builder 下面,选择非代理程序远程部署捆绑包。
- 3. 单击下一步。
- 4. 在项目名称字段中,输入项目名称。
- 5. 单击下一步。
- 6. 在"远程部署捆绑包信息"窗口中填写信息:
  - a) 在**捆绑包标识**字段中键入标识,此标识为 3 31 个字符的唯一字母数字字符串。此字符串可以包含连字符。此字符串必须以字母开头,但不能以 K 或连字符开头。
  - b) 在捆绑包描述字段中, 键入捆绑包描述。
  - c) 在**版本**字段中, 键入 VVRRMMFFF 格式的捆绑包版本。其中, vv 为版本号, rr 为发行版号, mm 为修 改号(修订包号), 而 fff 为临时修订号。
- 7. 在操作系统区域中,选择此捆绑包可以部署到的操作系统。
- 8. 单击完成以便在工作空间中创建项目,并打开远程部署捆绑包编辑器。

## 远程部署捆绑包编辑器

远程部署捆绑包编辑器用于生成帮助部署文件捆绑包的命令。

远程部署捆绑包编辑器提供有关项目捆绑包的信息。

捆绑包标识信息部分包含以下信息:

#### **Bundle identifier**

捆绑包的唯一标识

Bundle description 捆绑包的描述

#### **Bundle version**

捆绑包的版本

#### Build

捆绑包的构建标识。此处输入构建号。如果未指定任何构建号,那么将根据生成捆绑包的日期和时间生成一个编号。

针对捆绑包中的文件创建复制命令复选框

单击该复选框以生成在部署捆绑包时运行的一组缺省复制命令。这些文件将复制到复制位置文本框中指定的位置。缺省位置是 INSTALLDIR。在命令行部署中进行以下设置来指定此远程部署变量: KDY.INSTALLDIR=...

操作系统部分显示可以将捆绑包部署到的操作系统。

命令部分显示部署捆绑包时要运行的命令。

必备捆绑包部分显示为使此捆绑包工作而必须存在的捆绑包。

使用远程部署捆绑包编辑器来选择一组缺省复制命令,以便将捆绑包中的文件复制到指定位置。如果选择了此选项,那么会为捆绑包项目中的各个文件生成一个复制命令。缺省复制位置为 *INSTALLDIR*。如果未在部署命令行中设置特殊远程部署变量,那么缺省为 *CANDLEHOME*。要更改 *INSTALLDIR* 指定的位置,请在运行 addSystem 命令时指定 KDY.INSTALLDIR 属性。

捆绑包项目中指定的相同目录结构将复制到 *INSTALLDIR*。例如,如果捆绑包项目中名为 config 的文件夹 有一个名为 myprod.config 的文件,那么生成的复制命令会在部署捆绑包时将该文件复制到 *INSTALLDIR*/config/myprod.config。

# 向捆绑包添加命令

您可以指定更多要在部署期间运行的命令。

#### 关于此任务

通过使用**远程部署捆绑包编辑器**,您可以指定要在部署期间运行的其他命令。

#### 过程

- 1. 要指定部署期间运行的其他命令,请在远程部署捆绑包编辑器的命令部分中单击添加。
- 2. 在"命令"窗口中,选择命令类型预安装、安装、后安装或卸载,然后指定要运行的命令。

必须指定要运行的命令的标准路径。方便起见,远程部署提供了一组预定义变量。要在命令中引用变量,请在变量两侧加竖线,例如 | DEPLOYDIR |。有关命令的预定义变量的更多信息,请参阅<u>第 314 页的表 51</u>。

表 51. 命令的预定义变量	
变量	描述
DEPLOYDIR	这是端点上的一个临时目录,在部署期间,捆绑包存储在此目录中。例如,如果要运行捆绑包中包含的 myscript.sh 脚本,可以指定以下命令:  DEPLOYDIR /myscript.sh
INSTALLDIR	<i>CANDLEHOME</i> , 或者 <i>KDY</i> . <i>INSTALLDIR</i> 的值(如 果已在 addSystem 命令中指定)。
CANDLEHOME	Tivoli Monitoring 安装目录。

3. 最后,选择要在其中运行命令的系统。

# 向捆绑包添加必备项

使用远程部署捆绑包编辑器可以指定捆绑包的必备项。

## 过程

1. 要添加必备项,请在远程部署捆绑包编辑器的"捆绑包信息"页面的必备捆绑包部分中单击添加。

- 2. 在"新建必备项"窗口中,输入此捆绑包所依赖的捆绑包标识以及所需的最低版本。
- 3. 选择需要此必备项的操作系统。
- 4. 单击确定以完成操作并退出。

# 向捆绑包添加文件

通过使用远程部署捆绑包编辑器,可以向文件捆绑包添加文件。

#### 过程

- 1. 要将文件添加至远程部署捆绑包, 请执行下列其中一个过程:
  - · 在捆绑包编辑器中, 单击将文件添加至捆绑包。
  - · 在导航器树中,右键单击项目,然后单击 IBM Tivoli Monitoring 远程部署 > 将文件添加至捆绑包

这两项操作都将显示"导入捆绑包文件"窗口:

- 2. 在文件信息区域中,指定个别文件或包含文件的目录。
- 3. 单击完成。

所指定的文件或目录将复制到项目目录中。您构建远程部署捆绑包时,项目中的目录结构保持不变。如果要让 Agent Builder 生成缺省的复制命令,请确保文件所处的目录结构正确,可以进行部署。

# 生成捆绑包

使用 Agent Builder 为代理程序远程部署生成捆绑包。

#### 过程

1. 要生成远程部署捆绑包, 请使用以下其中一个过程来显示"生成最终远程部署捆绑包"窗口。

- · 在远程部署捆绑包编辑器中, 单击生成最终远程部署捆绑包。
- · 在导航器树中,右键单击项目,然后单击 IBM Tivoli Monitoring 远程部署 > 生成远程部署捆绑包。
- 2. 现在,您可以用以下两种方法生成捆绑包:
  - ·如果运行 Agent Builder 的系统上存在 Tivoli Enterprise Monitoring Server,请单击将远程部署捆绑包 安装到本地 TEMS 库中。

Agent Builder 尝试确定 Tivoli Monitoring 安装位置并将其输入到目录字段中。如果未设置 CANDLE\_HOME,将使用缺省位置 C:\IBM\ITM 或 /opt/ibm/itm。请确保安装位置正确,然后再继 续操作。

必须提供 Tivoli Enterprise Monitoring Server 登录信息才能安装捆绑包。

· 要将捆绑包生成到系统上的目录中,请单击在本地目录中生成远程部署捆绑包。

此过程完成之后,必须将此目录传输到 Tivoli Enterprise Monitoring Server 系统,并使用 tacmd addbundles 命令将此捆绑包添加到库。

#### 下一步做什么

在部署捆绑包时,必须使用 tacmd addSystem 命令。例如:

tacmd addsystem -t MONITORINGCOLLECTION -n Primary:ITMAGT:NT

其中-t(type)是下面的tacmd viewDepot命令返回的产品代码:

>tacmd viewDepot
Product Code : MONITORINGCOLLECTION
Version : 010000003
Description : MonitoringCollectionScripts
Host Type : WINNT
Host Version : WINNT
Prerequisites:

注: 您无法从 Tivoli Enterprise Portal Desktop 或 Browser 远程部署。从 Tivoli Enterprise Portal Desktop 或 Browser 远程部署会产生 KFWITM219E 消息。

请参阅 Tivoli Monitoring 文档以了解更多详细信息。

# 为 Tivoli Netcool/OMNIbus 探测器创建可部署捆绑包

您可以使用 Agent Builder 来创建包和配置捆绑包,这些包及配置捆绑包可用来将 Tivoli Netcool/OMNIbus 探测器部署到远程计算机。

## 关于此任务

为了支持探测器远程部署,在部署探测器之前,您还可以创建可部署至远程计算机的 Tivoli Netcool/OMNIbus 捆绑包。

## 过程

- 1. 从 Agent Builder 中,选择文件 > 新建 > 其他。
- 2. 在 IBM Tivoli OMNIbus 向导下面,选择软件包捆绑。
- 3. 单击**下一步**。

## 下一步做什么

接下来,使用 OMNIbus 安装捆绑包向导来创建捆绑包。有关使用此向导的信息,请参阅 <u>Tivoli Netcool/</u> OMNIbus 文档。

# 附录 H 动态文件名支持

使用动态文件名支持来指定文件名模式而非实际文件名。

某些应用程序可以创建遵从更改的输出文件名。名称根据当前日期、月份、年份或包含递增序号的文件名等 特定条件进行更改。在这些情况下,您可以指定文件名模式而非实际文件名。在指定文件名模式时,有两种 可以识别的模式格式:

·正则表达式(首选)。

·IBM Tivoli Universal Agent 动态文件名语法(不推荐)。

#### 正则表达式文件名模式

要指定文件名模式,您可以根据(<u>第309页的『附录 FICU 正则表达式</u>)中所记录的 International Components for Unicode (ICU) 语法来使用正则表达式。要使用此功能,您必须在"高级日志文件属性组信息"页面上选中文件名与正则表达式匹配复选框。在指定正则表达式模式时,还必须从"高级日志文件属性组信息"页面上的当多个文件匹配时列表中选择选项,以指定用于选择最新匹配文件的准则。

注:正则表达式是指定文件名模式的首选方法。

有关如何配置高级日志文件属性组属性的更多信息,请参阅(<u>第88页的『监视日志文件』</u>)中的步骤(<u>第89页的『6</u>」)。例如,如果指定了文件名模式:

d:\program files\logs\tivoli.\*

此模式将搜索 d:\program files\logs 目录中以 tivoli 开头的文件名。只能对文件名部分而不是路径 名指定正则表达式。

## 动态文件名语法

如果使用动态文件名语法,那么一次只能监视一个文件。文件数据提供程序检查所指定的路径位置中的所有 文件,查找与所定义的模式相匹配的文件。文件数据提供程序始终监视最新的匹配文件,判断依据是具有最 大数字或日期时间值的匹配文件名。要监视的相应文件由文件名确定,而不是由文件创建或其他条件确定。

可以指定文件名的模式,其可以具有任意数量的组成部分。例如,Log{###} 与只含有一个部分的文件名 (例如 Log010 或 Log456)匹配。在含有多个部分的文件名中,可以在文件名的任何部分或多个部分中指 定模式字符。例如,aaa.bbb{???}.ccc 是有效模式,aaa.bbb{???}.ccc{####} 也有效。

**注:** 指定文件名模式的首选方法是正则表达式,而不是动态文件名语法,有关正则表达式的更多信息,请参阅<u>第 317 页的『正则表达式文件名模式』</u>

以下示例说明了文件名模式规范:

#### 

与长度为 8 且文件扩展名为.abc 的数字文件名相匹配,例如 10252006.abc 或 10262006.abc。将 监视文件 10262006.abc,因为 10262006 大于 10252006。

#### *{########}*.\*

与长度为 8 的数字文件名相匹配,并忽略文件扩展名。示例包括 20061025.log、20061101.log 和 10252006.abc。将监视文件 20061101.log,因为 20061101 是最大数字。

#### {######??}.abc

与长度为 8 且文件扩展名为 .abc 的数字文件名相匹配,并忽略名称部分的最后两位。示例包括 02110199.abc、02110200.abc 和 021101AZ.abc。将监视文件 02110200.abc,因为 021102 是 最大数字。

#### Console. {#######}

与名称部分包含 *Console* 且扩展名为六位数字的文件名相匹配。示例包括 Console.000133、 Console.000201 和 Console.000134。将监视文件 Console.000201。

#### IN{######}.log

与以 IN 开头, 后跟 6 个数字, 且文件扩展名为 .log 的文件名相匹配。示例包括 INO21001.log、 INO21002.log 和 INO21004.log。将监视文件 INO21004.log。

#### PS{###}FTP.txt

与以 PS 开头,后跟 3 个数字,然后再跟 FTP, 且扩展名为.txt 的文件名相匹配。示例包括 PS001FTP.txt、PS005FTP.txt 和 PS010FTP.txt。将监视文件 PS010FTP.txt。

请遵循以下准则来确立文件名模式:

·使用花括号 {} 括起文件名中的模式字符。花括号内存在模式字符表示使用了文件名模式。

·使用星号 (\*) 作为通配符以忽略文件扩展名或文件名中的任何尾随字符。例如, Myapp {###}.log\* 指定以 Myapp 开头, 后跟三个数字, 然后再跟.log(无论后面再跟什么字符)的任何文件名都是匹配项。

星号必须在花括号({ })后面指定,而不能用于文件名开头。在文件扩展名中使用星号时,星号必须独立使用。

通配符(\*)正确用法示例:

err{??}.\*

## error{\$}.\*

通配符 (\*) 错误用法示例:

#### error.20\*

星号(\*)之前没有花括号。

error\*.{###}

星号不能在文件名结尾使用。

error.\*

星号(\*)之前没有花括号。

·如果定义了特定的文件扩展名,那么只考虑具有相同扩展名的文件。

- ·使用井号来表示文件名的每个数字元素。
- ·使用问号来排除命名约定中不作为搜索条件来决定合适文件名的各个元素。
- ·使用美元符号(\$)来表示任意字符或没有字符。例如,如果要匹配名为Log和LogA的两个文件,请指定Log{\$}。美元符号有几个使用限制。在使用一个或多个美元符号作为文件名前缀时(例如 {\$\$\$\$ \$\_abc.log),美元符号的数量必须与文件名中该位置处的字符数完全相等。另外,也不能在文件名模式中的多个位置指定美元符号,例如 {\$\$\$}b{\$\$\$.log将不能匹配abc.log。由于存在这些美元符号限制,因此如果文件名中的字符数量不确定,请使用正则表达式文件名模式。
- ·花括号括起的井号和问号的总数很重要。它必须与文件名的部分完全匹配。例如,模式 AA {#####}指示文件数据提供程序查找类似 AA0001 的文件名。将不考虑诸如 AA001 或 AA00001 之类的文件名。
- ·精确文件名模式、常量和数字部分必须与文件名完全匹配。例如,模式 AA {###} 指示文件数据提供程序检查文件 AA101。将不考虑诸如 XAA101、AA222X 和 AA55555 之类的文件名。
- ·使用保留的模式字符串 {TIVOLILOGTIME} 来替换 Tivoli Monitoring 代理程序或服务器日志文件中的十六 进制时间戳记和文件序列号。在对 Tivoli Monitoring 组件实施自监视时,此模式字符串很有用。例如,如 果要监视 /opt/IBM/ITM/logs 目录中的最新 监视服务器 日志,可以指定文件名模式:

/opt/IBM/ITM/logs/Host1\_ms\_{TIVOLILOGTIME}.log

如果 /logs 目录中存在 Host1\_ms\_452053c0-01.log、Host1\_ms\_451f11f4-01.log、 Host1\_ms\_45205946-01.log 和 Host1\_ms\_451f11f4-02.log,将选择监视 Host1\_ms\_45205946-01.log 文件。

要精确指定由各个日期部分(年、月和日)组成的文件名,使用大写字母 Y、M 和 D。必须在括号内指定这些字母;否则,它们将被视为文件名中的文字字符。

请参阅以下示例:

#### {YYYYMMDD}.log

指定诸如 20060930.log 或 20061015.log 之类的文件名。

#### {MMDDYY}.log

指定诸如 101106.log 或 110106.log 之类的文件名。

#### {DDMMYYYY}.log

指定诸如 01092006.log 或 15082006.log 之类的文件名。

#### {DDMMMYY}.log

指定诸如 24Jan07 或 13Sep06 之类的文件名。

#### {MM-DD-YY}.log

指定诸如 11-02-06 或 04-29-07 之类的文件名。日期字段中的 (-) 分隔符会被忽略,因此不需要使用 问号模式字符跳过此字符。

#### MY{YYDDD}.log

指定诸如 MY06202.log、MY06010.log 或 MY04350.log 之类的文件名。

文件数据提供程序会定期检查目标路径位置中与定义的文件模式匹配的新文件。当检测到与模式匹配的较新 文件时,文件数据提供程序自动将应用程序监视切换至该新文件。在以下情况下,文件数据提供程序会搜索 最佳匹配文件:

·文件数据提供程序第一次启动时。

- ·由于可能发生重命名或删除,当前受监视的文件不再存在。
- ·现有文件内容由于可能发生重写而已更改。
- ·检查时间间隔已到期。缺省时间间隔为10分钟。您可以通过指定环境变量来增大或减小时间间隔

KUMP\_DP\_FILE\_SWITCH\_CHECK\_INTERVAL=number-of-seconds

# 附录 I SNMP 陷阱配置

这是对配置文件的描述, SNMP 数据提供程序使用此配置文件以更容易阅读的形式呈现陷阱信息。该文件还用于向陷阱分配类别、严重性、状态和源标识。

此外还包含有关如何修改缺省文件或替换您自己的配置文件的指示信息。

## SNMP 陷阱配置文件 trapcnfg

在启动时,SNMP 数据提供程序读取名为 trapcnfg 的配置文件。该文件的一个用途是将 SNMP 陷阱信息转换为更容易阅读的形式。另一个用途是向特定陷阱分配类别、严重性、状态和源标识,因为 SNMP 未定义这些类别。

您可以通过添加新的陷阱或企业定义或更改现有的陷阱或企业定义来修改 trapcnfg 文件以满足特定于站点的需要。还可以使用自己的配置文件。

#### 使用 HP OpenView trapd.conf 文件

trapcnfg 文件与 HP OpenView Network Node Manager 陷阱配置文件 trapd.conf 在格式上相似,但并不完全相同。您可以在必要的情况下复制 OpenView 文件并复用许多定义语句。

#### 记录类型

trapcnfg 包含三种类型的记录或记录块:

注释

注释记录以数字符号(#)开头。

企业定义

企业定义由两个空白定界标记组成,其中第一个标记是名称;第二个标记是两边为大括号 ({ }) 的对象标 识 (OID)。

陷阱定义

陷阱定义由8个空白定界标记组成。陷阱定义是块记录,因为每个定义可能由多条记录构成。

第一种类型是自我解释型。(第 322 页的图 57)显示了第二种和第三种类型的示例。

<u>第 322 页的图 57</u>中第一个示例显示的企业定义记录将企业 OID 1.3.6.1.4.1.311.1.1.3.1.1 定义为 Microsoft Windows NT。

第二个示例显示的陷阱定义记录将 trapName MSNTCOLD 定义成与企业 OID

1.3.6.1.4.1.311.1.1.3.1.1 关联的一般陷阱号 0 和特定陷阱号 0。请注意,严重性是以十进制形式表示,而类别是以文本形式表示。严重性在显示之前转换为其文本形式。类型 3 记录块中的下一条记录是 Agent Builder 不使用的简短描述。Agent Builder 使用定界符 SDESC 和 EDESC 之间括起的详细描述。



A coldStart toop signifies that the sending protocol entry is reinitializing itself in such a way that the agent sconfiguration or the protocol entry implementation may be altered.

EDESC

图 57. 第二和第三种类型的配置记录示例

#### trapcnfg 文件的缺省值

以下各表列出了 SNMP 数据提供程序支持的缺省值。

#### 受支持的类别

(<u>第 322 页的表 52</u>)显示了 Agent Builder 支持的类别。

表 52. SNMP 数据提供程序支持的类别	
类别	文本说明
0	阈值事件
1	网络拓扑事件
2	错误事件
3	状态事件
4	节点配置事件
5	应用程序警报事件
6	所有类别事件
7	仅日志事件
8	映射事件
9	忽略事件

(<u>第 323 页的表 53</u>)列出了 Agent Builder 支持的严重性。

表 53. SNMP 数据提供程序支持的严重性	
严重性	文本说明
0	正常
1	不确定
2	警告
3	轻微错误
4	严重
5	主要错误

# 受支持的状态

(<u>第 323 页的表 54</u>)显示 Agent Builder 配置文件中定义的状态。

表 54. SNMP 数据提供程序支持的状态	
状态	文本说明
0	不变
1	未知
2	运行
3	不佳
4	关闭
5	未受管
6	确认
7	用户 1
8	用户 2

# 受支持的源标识

(<u>第 323 页的表 55</u>)列出了 trapcnfg 支持的源标识。

表 55. SNMP 数据提供程序支持的源标识	
源标识	描述
a	应用程序
А	代理程序
С	Xnmcollect
d	演示
D	数据收集器
E	Nvevents
如果	Ipmap
L	LoadMIB
m	Shpmon
М	IP 拓扑

表 55. SNMP 数据提供程序支持的源标识 (续)	
源标识	描述
n	netmon 相关
Ν	netmon 生成的陷阱
0	OSI SA
Р	非 IP 陷阱
r	Tralertd
S	Spappld
S	安全代理程序
t	Xnmtrap
т	Trapd
V	供应商相关
?	未知

# 附录 J"执行操作"命令参考

"执行操作"命令概述、有关"执行操作"命令的参考资料以及特殊"执行操作"命令的描述。

#### 关于"执行操作"命令

"执行操作"命令可包括在 Agent Builder 监视代理程序中。"执行操作"命令可以从门户网站客户机运行,或者包括在情境或策略中。包括在情境中时,这些命令会在该情境变为 true 时运行。情境中的"执行操作"命令也称作反射自动化。在情境中启用"执行操作"命令后,就会自动响应系统状况。例如,可以使用"执行操作"命令来发送一条命令,以便在受管系统上重新启动进程。另外,还可以使用"执行操作"命令向手机发送短信。

高级自动化使用策略来运行操作、调度工作以及自动完成手动任务。策略由一系列的自动化步骤组成,这些步骤称为活动,它们相互连接以构成工作流程。完成一项活动之后,Tivoli Enterprise Portal 会收到返回码反馈,高级自动化逻辑会以该反馈所指定的后续活动进行响应。

基本"执行操作"命令会在操作完成后显示的消息框或日志文件中显示操作返回码。关闭此窗口后,不会提供有关此操作的更多信息。

#### 关于"执行操作"命令的更多信息

有关使用"执行操作"命令的更多信息,请参阅《Tivoli Enterprise Portal 用户指南》。

要获取此监视代理程序的"执行操作"命令的列表和描述,请参阅<u>第325页的『特殊"执行操作"命令』</u>。 另请参阅该章节中每个命令的信息。

#### 特殊"执行操作"命令

Agent Builder 监视代理程序可以识别一组"执行操作"命令,并为其执行特殊处理:

· SSHEXEC

有关创建这些命令以及将它们包括在 Agent Builder 监视代理程序项目中的更多信息,请参阅<u>第181页的</u> <u>『第11章 创建工作空间、"执行操作"命令和情境』</u>。

## SSHEXEC 操作

#### 开始之前

有关"执行操作"命令的更多信息,请参阅第 325 页的『附录 J "执行操作"命令参考』。

#### 关于此任务

SSHEXEC 操作是针对至少有一个 SSH 脚本属性组的受监视应用程序而识别。它表示 SSHEXEC 关键字后的 命令要在 SSH 目标系统上远程启动。该命令将会使用为监视 SSH 目标系统而配置的用户的凭证和特权来启 动。该命令将在"受管系统名称"所表示的远程系统上运行。

## 过程

要将"执行操作"命令包括在情境或工作流策略中,请使用系统命令的以下语法:

SSHEXEC [Command]

例如:

SSHEXEC [1s &path]

**注:** 在调用"执行操作"期间,通过将"执行操作"自变量选项与 Command 配合使用,可以对命令或命令中的某些部分进行定制。

注:如果 Command 包含多个自变量,请考虑添加方括号,以支持通过 tacmd 命令行界面调用"执行操作" 命令。

# <u>辅助功能</u>

辅助功能帮助那些身体残障(例如行动有障碍或视力有障碍)的用户成功地使用信息技术内容。

## 辅助功能

IBM Cloud Application Performance Management 基于 Web 的界面是 Cloud APM 控制台。该控制台包含以下主要的辅助功能:

- ·使用户能够使用辅助技术(例如屏幕朗读器软件和数字发音合成器)来听取屏幕上显示的内容。有关将这些技术用于本产品的详细信息,请查阅辅助技术的产品文档。
- ·使用户能够仅使用键盘操作特定的或等效的功能部件。
- ·传达与颜色无关的所有信息。1

Cloud APM 控制台使用最新的 W3C 标准 WAI-ARIA 1.0 (http://www.w3.org/TR/wai-aria/),以确保符合 US Section 508 (http://www.access-board.gov/guidelines-and-standards/communications-and-it/about-the-section-508-standards/section-508-standards)和 Web Content Accessibility Guidelines (WCAG) 2.0。要利用辅助功能,请结合使用最新发行版的屏幕朗读器与该产品支持的最新 Web 浏览器。

IBM Knowledge Center 中的 Cloud APM 控制台联机产品文档支持辅助功能选项。IBM Knowledge Center 的辅助功能在 IBM Knowledge Center 发行说明中进行了描述.

## 键盘导航

本产品使用标准的导航键。

## 界面信息

Cloud APM 控制台 Web 用户界面不依靠级联样式表来正确呈现内容和提供适用的体验。但是,产品文档依 靠级联样式表。IBM Knowledge Center 为视力低下用户提供了等效方法来使用其定制显示设置,包括高对 比度模式。您可以使用设备或浏览器设置来控制字体大小。

Cloud APM 控制台 Web 用户界面包括您可以用于快速导航到应用程序中的功能性区域的 WAI-ARIA 导航标志。

Cloud APM 控制台用户界面不包含每秒闪烁 2 - 55 次的内容。

## 相关辅助功能选项信息

除了标准的 IBM 帮助热线和支持 Web 站点, IBM 还建立了 TTY 电话服务,以供耳聋或由严重听力障碍的客 户用于获取销售和支持服务:

TTY 服务 800-IBM-3383 (800-426-3383)(北美地区)

## IBM 与辅助功能

有关 IBM 对辅助功能选项所作承诺的更多信息,请参阅 IBM Accessibility (www.ibm.com/able)。

<sup>&</sup>lt;sup>1</sup> 但 Performance Management 控制台的某些代理程序配置页面除外。

声明

本信息是为在美国国内供应的产品和服务而编写的。IBM 可能在其他国家或地区不提供本文档中讨论的产品、服务或功能特性。有关您当前所在区域的产品和服务的信息,请向您当地的 IBM 代表咨询。任何对 IBM 产品、程序或服务的引用并非意在明示或暗示只能使用 IBM 的产品、程序或服务。只要不侵犯 IBM 的知识产权,任何同等功能的产品、程序或服务,都可以代替 IBM 产品、程序或服务。但是,评估和验证任何 非 IBM 产品、程序或服务,则由用户自行负责。

IBM 公司可能已拥有或正在申请与本文档内容有关的各项专利。提供本文档并未授予用户使用这些专利的任何许可。您可以用书面方式将许可查询寄往:

IBM Director of Licensing IBM Corporation North Castle Drive Armonk, NY 10504-1785 U.S.A.

有关双字节 (DBCS) 信息的许可查询,请与您所在国家或地区的 IBM 知识产权部门联系,或用书面方式将查询寄往:

Intellectual Property Licensing Legal and Intellectual Property Law IBM Japan, Ltd. 19-21, Nihonbashi-Hakozakicho, Chuo-ku Tokyo 103-8510, Japan

#### 本条款不适用英国或任何这样的条款与当地法律不一致的国家或地区:

International Business Machines Corporation "按现状"提供本出版物,不附有任何种类的(无论是明示的 还是默示的)保证,包括但不限于默示的有关不侵权、适销和适用于某种特定用途的保证。

某些国家或地区在某些交易中不允许免除明示或默示的保证。因此本条款可能不适用于您。

本信息中可能包含技术方面不够准确的地方或印刷错误。此处的信息将定期更改;这些更改将编入本资料的 新版本中。IBM 可以随时对本出版物中描述的产品进行改进和/或更改,而不另行通知。

本信息中对非 IBM Web 站点的任何引用都只是为了方便起见才提供的,不以任何方式充当对那些 Web 站点的保证。那些 Web 站点中的资料不是 IBM 产品资料的一部分,使用那些 Web 站点带来的风险将由您自行承担。

IBM 可以按它认为适当的任何方式使用或分发您所提供的任何信息而无须对您承担任何责任。

本程序的被许可方如果要了解有关程序的信息以达到如下目的: (i) 允许在独立创建的程序和其他程序(包括本程序)之间进行信息交换,以及(ii) 允许对已经交换的信息进行相互使用,请与下列地址联系:

IBM Corporation 2Z4A/101 11400 Burnet Road Austin, TX 78758 U.S.A.

只要遵守适当的条件和条款,包括某些情形下的一定数量的付费,都可获得这方面的信息。

本资料中描述的许可程序及其所有可用的许可资料均由 IBM 依据 IBM 客户协议、IBM 国际软件许可协议或 任何同等协议中的条款提供。

此处包含的任何性能数据都是在受控环境中测得的。因此,在其他操作环境中获得的数据可能会有明显的不同。有些测量可能是在开发级的系统上进行的,因此不保证与一般可用系统上进行的测量结果相同。此外,有些测量是通过推算而估计的,实际结果可能会有差异。本文档的用户应当验证其特定环境的适用数据。

涉及非 IBM 产品的信息可从这些产品的供应商、其出版说明或其他可公开获得的资料中获取。IBM 没有对 这些产品进行测试,也无法确认其性能的精确性、兼容性或任何其他关于非 IBM 产品的声明。有关非 IBM 产品性能的问题应当向这些产品的供应商提出。 所有关于 IBM 未来方向或意向的声明都可随时更改或收回, 而不另行通知, 它们仅仅表示了目标和意愿而已。

版权许可:

本信息包括源语言形式的样本应用程序,这些样本说明不同操作平台上的编程方法。如果是为按照在编写样本程序的操作平台上的应用程序编程接口 (API) 进行应用程序的开发、使用、经销或分发为目的,您可以任何形式对这些样本程序进行复制、修改、分发,而无须向 IBM 付费。这些示例并未在所有条件下作全面测试。因此, IBM 不能担保或默示这些程序的可靠性、可维护性或功能。用户如果是为了按照 IBM 应用程序编程接口开发、使用、经销或分发应用程序,则可以任何形式复制、修改和分发这些样本程序,而无须向 IBM 付费。

如果您正在查看本信息的软拷贝,图片和彩色图例可能无法显示。

# 商标

IBM、IBM 徽标和 ibm.com<sup>®</sup> 是 International Business Machines Corp. 在全球许多管辖区域的商标或注册 商标。其他产品和服务名称可能是 IBM 或其他公司的商标。IBM 商标的当前列表可在 Web 上找到,其网址 为 <u>http://www.ibm.com/legal/us/en/copytrade.shtml</u>。

Adobe、Adobe 徽标、PostScript 和 PostScript 徽标是 Adobe Systems Incorporated 在美国和/或其他国家 或地区的注册商标或商标。

Linux 是 Linus Torvalds 在美国和/或其他国家或地区的商标。



Java 和所有基于 Java 的商标和徽标是 Oracle 和/或其附属公司的商标或注册商标。

Microsoft 和 Windows 徽标是 Microsoft Corporation 在美国和/或其他国家或地区的注册商标。

UNIX 是 The Open Group 在美国和/或其他国家或地区的注册商标。

其他公司、产品和服务名称可能是其他公司的商标或服务标记。

